TECHNISCHE
UNIVERSITÄT
DARMSTADT

COLLABORATIVE SENSING IN AUTOMOTIVE SCENARIOS

Enhancement of the Vehicular Electronic Horizon through Collaboratively Sensed Knowledge

Vom Fachbereich Elektrotechnik und Informationstechnik
der Technischen Universität Darmstadt
zur Erlangung des akademischen Grades eines
Doktor-Ingenieurs (Dr.-Ing.)
genehmigte Dissertation

von

DANIEL-MAURICE BURGSTAHLER, M.SC.

Geboren am 23. Mai 1980 in Wiesbaden

Vorsitz: Prof. Dr. Mario Kupnik
Referent: Prof. Dr.-Ing. Ralf Steinmetz
Korreferent: Prof. Dr.-Ing. Klaus David

Tag der Einreichung: 14. Februar 2017
Tag der Disputation: 19. Juni 2017

Hochschulkennziffer D17
Darmstadt 2017

ABSTRACT

Modern vehicles are equipped with a variety of advanced driver assistance systems that increase driving comfort, economy and safety. Respective information sources for these systems are local sensors, like cameras, radar or lidar. However, the next generation of assistant systems will require information above the local sensing range. An extension of the local perception can be provided by the use of appropriate communication mechanisms. Hence, other vehicles can serve as an information source by providing their local perception data, but also any other information source, such as cloud services. Required communication can take place directly between vehicles via mobile ad-hoc communication or via a backend by the use of cellular communication. The appropriate technology depends on the respective use case, that determines information content, granularity and tolerated latency. Based on literature, we derived a categorization of use case dependent information demands, with respect to communication. The resulting three zones, namely *safety zone*, *awareness zone* and *information zone*, refer to the tolerated latency between the occurrence of an information and the point in time the information has to be processed at the receiver side. While communication mechanisms for the *safety zone*, i. e., the ego-vehicle's direct surroundings with a remaining driving time of less than $2-5$ seconds, have been focus in research and standardization in the past, respective mechanisms for larger distances have not been sufficiently considered. In this thesis, we examine information distribution mechanisms in context of the previously mentioned use case categories.

As the first key contribution, we consider the gathering of vehicular sensed data with regard to the *information zone*, i. e., more than 30 seconds remaining driving time to the point of the information origin. We developed a probabilistic data collection model that is able to reduce data traffic up to 85 % compared to opportunistic transmission and still sticks to certain quality metrics, e. g., a maximum detection latency. A central adaption of transmission probabilities to the density of transmitting vehicles is applicable for cellular use and copes with sparse traffic situations. Moreover, we have extended this approach by hybrid communication, i. e., the parallel use of cellular and mobile ad-hoc communication. This allows to further reduce cellular based data traffic, in particular in case of dense traffic.

As the second key contribution, we examine the efficient distribution of the previously gathered information. Information is structured and prioritized according to the most probable driving path, as so-called electronic horizon. The transmission towards the vehicles is performed in small data packets, according to the given priorities. The aim is to transmit only information relevant for road segments that will be used. Concerning this, we developed a mechanism for most probable travel path estimation and a data structure for efficient mapping of the electronic horizon.

As the third key contribution, we examine the information exchange in the *awareness zone*, an area between the *safety zone* and the *information zone* with about 5 to 30 seconds remaining driving time to the point of the information origin. Derived from the respective use cases, this data is not directly safety relevant, but it is still about dynamic position information of neighboring vehicles. Due to the relatively

long distance, direct vehicle to vehicle communication is not possible. Respective data has to be forwarded by intermediate vehicles. However, position beacons without data forwarding can already cause channel congestion in dense traffic situations. The use of cellular networks would require absolute total network coverage with permanent free channel resources. To enable forwarding of dynamic vehicle information anyhow, we developed at first a mechanism to reduce the channel load for position beacons. Next, we use the freed-up bandwidth to forward dynamic information about neighboring vehicle positions. With this mechanism, we are able to more than double the range of vehicular perception, with respect to moving objects.

In extension to standardized communication mechanisms for the safety relevant direct proximity, our three mentioned contributions provide the means to complete the long range vehicular perception for future advanced driver assistance systems.

KURZFASSUNG

Moderne Fahrzeuge sind mit einer Vielzahl von Fahrerassistenzsystemen ausgestattet, welche das Fahren in Bezug auf Komfort, Wirtschaftlichkeit und Sicherheit verbessern. Informationsbasis sind hierfür Sensorinformationen lokal verbauter Sensoren, wie beispielsweise Kameras, Radarsensoren oder Lidar. Die nächste Stufe von Fahrerassistenzsystemen benötigt Informationen über die Sichtweite lokal verbauter Sensoren hinaus. Diese Informationen können mittels geeigneter Kommunikationsmechanismen bereitgestellt werden. Zur Erfassung dieser Informationen können andere Fahrzeuge als mobile Sensorknoten dienen, aber auch beliebige andere Informationsquellen, wie beispielsweise Cloud Dienste. Die Kommunikation kann direkt zwischen den Fahrzeugen oder über ein Backend erfolgen. Als Kommunikationstechnologie kann hierbei Mobilfunk, aber auch direkte Kommunikation zwischen einzelnen Fahrzeugen eingesetzt werden. Welcher Kommunikationsweg und entsprechender Kommunikationsmechanismus geeignet ist, hängt vom jeweiligen Anwendungsfall ab, welcher den Informationsinhalt, die Granularität der Daten und die tolerierbare Latenz bestimmt. Dazu wurde aus der Literatur eine Klassifikation von Anwendungsfällen bezüglich der Kommunikationsanforderungen abgeleitet. Die daraus resultierenden Bereiche *Sicherheitszone*, *Achtsamkeitszone* und *Informationszone* beziehen sich auf die tolerierbare Latenz zwischen dem Auftreten einer Information und der Verarbeitung auf der Empfängerseite. Kommunikationsmechanismen für die *Sicherheitszone*, also den direkten Nahbereich um das Ego-Fahrzeug mit einer Restfahrzeit bis zum Ereignispunkt von weniger als 2-5 Sekunden, wurden in den letzten Jahren intensiv in Forschung und Standardisierung behandelt. Im Gegensatz dazu wurden geeignete Informationsverteilmechanismen für entferntere Bereiche nicht hinreichend behandelt. In der vorliegenden Arbeit wurden hierzu geeignete Informationsverteilmechanismen im Kontext der zuvor genannten Anwendungsfälle untersucht.

Der erste Kernbeitrag dieser Arbeit befasst sich mit dem effizienten Erfassen von Informationen für den Bereich *Informationszone*, mit mehr als 30 Sekunden Restfahrzeit bis zum Ort der Information. Das entwickelte probabilistische Erfassungsmodell ermöglicht das Einhalten bestimmter Qualitätsmetriken, wie eine maximale Erkennungslatenz, bei gleichzeitiger Reduktion des Datenvolumens um bis zu 85% gegenüber einem opportunistischen Übertragungsmodell. Die zentrale Adaption der Übertragungswahrscheinlichkeiten an die Dichte der sendenden Fahrzeuge ist geeignet für eine mobilfunkbasierte Erfassung und auch für geringe Fahrzeugdichten anwendbar. Durch eine Erweiterung um hybride Kommunikation, d.h. die parallele Verwendung von Mobilfunk und mobiler Ad-hoc Kommunikation, kann eine weitere Verringerung des mobilfunkbasierten Datenvolumens gezeigt werden. Besonders in dichtem Verkehr bietet dies ein großes Einsparpotential.

Der zweite Kernbeitrag dieser Arbeit befasst sich mit dem effizienten Verteilen der zuvor gesammelten Informationen. Basierend auf dem wahrscheinlichsten Fahrpfad werden die Informationen als so genannter elektronischer Horizont aufgebaut und priorisiert. Die Übertragung zum Fahrzeug erfolgt in kleinen Paketen, gemäß den Prioritäten. Ziel ist es möglichst nur Informationen zu den Streckenabschnit-

ten zu übertragen, welche auch befahren werden. Dazu wurde ein Verfahren zur Abschätzung des Fahrpfades und eine Datenstruktur zur effizienten Abbildung des elektronischen Horizonts entwickelt.

Der dritte Kernbeitrag dieser Arbeit befasst sich mit dem Informationsaustausch in der *Achtsamkeitszone*, einem Bereich zwischen der *Sicherheitszone* und der *Informationszone* mit etwa 5 bis 30 Sekunden Restfahrzeit bis zum Ort der Information. Abgeleitet aus den Anwendungsfällen handelt es sich hierbei zwar nicht um sicherheitskritische Informationen, aber dennoch vorwiegend um dynamische Informationen zur Position von Nachbarfahrzeugen. Aufgrund der relativ großen Distanz ist eine direkte Fahrzeug-zu-Fahrzeug Kommunikation nicht möglich. Entsprechende Daten müssen hierfür von dazwischenfahrenden Fahrzeugen weitergeleitet werden. Bei dichtem Verkehr kann der Funkkanal durch die Positionsnachrichten der Fahrzeuge bereits ohne Weiterleitung vollständig ausgelastet sein. Eine entsprechende Realisierung über Mobilfunk würde eine absolut flächendeckende Netzabdeckung und stetige Kanalressourcen voraussetzen. Um dennoch eine Weiterleitung dynamischer Fahrzeuginformationen zu ermöglichen, wurde zunächst ein Verfahren zur Reduzierung der durch die Positionsnachrichten der Fahrzeuge erzeugten Kanallast entwickelt. Die so frei gewordene Kanalkapazität kann nun für die Weiterleitung von dynamischen Positionsinformationen benachbarter Fahrzeuge verwendet werden. Durch diesen Mechanismus kann die Reichweite des lokalen Wissens über dynamische Informationen von Nachbarfahrzeugen mehr als verdoppelt werden.

In Ergänzung zu den bereits standardisierten Kommunikationsmechanismen für den direkten sicherheitskritischen Nahbereich, ermöglichen die drei genannten Beiträge die Vervollständigung des für zukünftige Fahrerassistenzsystemen notwendigen Wahrnehmungsbereichs durch effiziente, anwendungsfallspezifische Kommunikation.

CONTENTS

1 INTRODUCTION

Cooperative Intelligent Transport System (C-ITS) technology has a significantly growing importance in the vehicular domain and already reaches back to the late 1970s [1]. While in those days information and communication technology was not sufficiently developed to realize the visions, today the situation has changed. Vehicular connectivity is fast growing and getting more and more relevant as an essential component of Advanced Driver Assistance Systems (ADAS). Communication technology can be used to increase knowledge about the local environment, the so called vehicular perception. However, research activities mainly address single hop information distribution. Research results on how to increase the long-range view of vehicular perception for application dependent quality demands, by the use of communication, will be addressed in this thesis.

In this chapter, we first outline the motivation for this work in Section 1.1, followed by a description of resulting challenges and an overview of our key contributions in Section 1.2. In Section 1.3, we conclude the chapter with an outline of the overall thesis.

1.1 MOTIVATION

The amount of vehicles is constantly increasing, with a total of more than 91 million produced vehicles in 2016 worldwide [2, 3, 4]. As a direct consequence, vehicular traffic is getting more dense and complex, with more accidents and a higher environmental impact. As a result, traffic efficiency is reduced and as a consequence thereof, overall economic. This forces the need of Advanced Driver Assistance Systems (ADASs), that intend to increase traffic efficiency and driving comfort, economy and safety. These systems require a reliable perception of the vehicle status and its surroundings, which is commonly realized with a set of built-in sensors [5]. Most important sensors for the vehicular perception are vision based camera systems, as well as RAdio Detection And Ranging (RADAR) and LIght Detection And Ranging (LIDAR) sensors. An example illustration of the vehicular local perception is depicted in Figure 1, whereas each color represents a different type of sensor. Typically, a vehicle has built-in more than one instance of each sensor type, to cover all required areas. However, the respective sensing range is limited and relatively short [6]. This limited perception range limits the capabilities of ADASs. To enhance existing ADAS and even enable completely new ADAS concepts, an extension of this perception range is required, that also enables more predictive systems. A possible extension of the perception range is the use of geo localization, i. e., by the use of a Global Navigation Satellite System (GNSS), combined with digital map data, to generate an electronic Horizon (eHorizon) (c.f. Chapter 2.3) as virtual predictive sensor [5]. It enables an extended forward view into the direction of the driving path, enriched with information about road geometry and attached attributes, e. g., road gradients or the position of speed limits. Such an extended perception range becomes

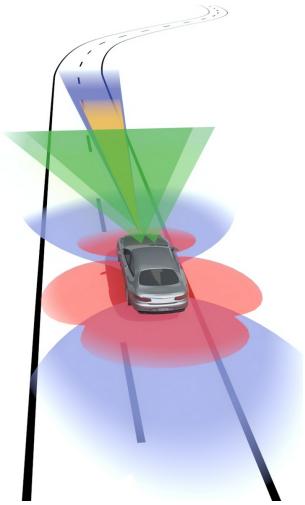

Figure 1: Illustration of the local perception of a vehicle equipped with sensors. Each color represents a different type of sensor.

in particular important, in case of *highly automated* and *fully automated driving* (c.f. Section 2.2). However, map data can be rapidly outdated, which would require frequent updates. But beside the map update problem, the acquisition of map relevant changes itself is a complex task. First of all, these changes are, with about 15% per year, much more than one would expect [7]. These changes not only include the road geometry itself, but also a change in traffic signs and rules. Moreover, the update process takes a relatively long time. Either updates are gathered by map vendors within explicit measurement drives or they are provided within a crowd sourced approach, that requires a validation process. An innovation could be a better update process for this type of information. Yet, future ADASs require not only up to date information about the road network itself, but also dynamic information about the vehicle surroundings, in particular with an increasing degree of automation [8].

To solve this issue, other vehicles can serve as an information resource, sharing locally sensed information that is potentially valuable for other vehicles [9]. The respective information exchange can be realized by the means of wireless communication technology, soon becoming standard equipment. Future vehicles will be equipped with communication units, capable to use different communication technologies in parallel [9]. This will allow the use of many new information sources in future ADASs. Basically, this communication can be structured into three categories, namely Vehicle to Vehicle (V2V), Vehicle to Infrastructure (V2I) and Vehicle to everything (V2X). V2V denotes direct inter-vehicle communication, realized via mobile ad-hoc communication, whereas neighboring vehicles exchange information about their current status and dynamics [10]. In V2I communications, vehicles communicate with intelligent road infrastructure, e. g., traffic lights, broadcasting the current signal status and signal phase timing information [11]. Finally, with V2X vehicles communicate with their smart and connected environment, driven by the development of smart cities and Internet of Things (IoT), that allows to incorporate detailed information into future ADASs, e. g., exact parking space information in single parking space granularity [12]. As a result, the amount of available information for future ADASs will be tremendous, which requires intelligent and use case dependent information distribution, that prevents overload of communication channels.

1.2 CHALLENGES AND CONTRIBUTIONS

In order to realize the above mentioned extension of the vehicular perception range, use case dependent information distribution strategies are required. Moreover, different communication technologies are available for realization, that bring different means and restrictions. In general, the vehicular information exchange can either be realized by infrastructure-based or direct V2V communication. Our aim is to provide the foundation for future ADASs by extended vehicular perception, based on wireless communication technology. This will enable to realize more predictive ADASs and better support in cooperative driving maneuvers. Hence, the main research questions addressed in this thesis are: *How to collect perception information efficiently?*, *Which information is relevant?* and *How to efficiently distribute respective information?*

Connecting vehicles with wireless communication technology has been focused in research since many years and has brought many concepts in the field of Vehicular Ad-hoc NETworks (VANETs). The main purpose is the improvement of ADASs, with respect to safety applications [13, 14]. The field is summarized in Europe under the term Cooperative Intelligent Transport System (C-ITS). In the US, the term Dedicated Short Range Communication (DSRC) is more common. Many solutions have been already transferred into standardization at IEEE, SAE (in the US) and ETSI (in Europe). For both, U.S. and Europe, communication is built on top of the IEEE 802.11p data link and physical layer in the 5.9 GHz range and seven 10 MHz channels have been assigned for vehicular communication. Most important applications are based on being aware of the neighbor vehicles' positions. Up to one complete channel is used to broadcast the respective position tracking relevant messages (Basic Safety Message (BSM) in the U.S and Cooperative Awareness Message (CAM) in Europe, more details are given in Section 2.4.2). This allows to enhance the local perception range, with respect to moving objects in the range of one communication hop. But several use cases, e. g., cooperative maneuvers, will need an even larger perception range. An alternative are cellular networks, that provide wide coverage and high data rates. But safety relevant use cases have high latency demands, that cannot always be guaranteed [15]. Moreover, the use of cellular networks is costly. In contrast, DSRC, i. e., 802.11p based vehicular mobile ad-hoc communication, can support real time communication, but has a limited range and channel capacity. The used Carrier Sense Multiple Access (CSMA) mechanism in the Medium Access Control (MAC) layer effects a high probability of packet collisions in case of dense traffic, which causes low channel utilization and an increased transmission latency [16]. In addition, DSRC might also get problems in case of sparse traffic situations, even with a high penetration rate of equipped vehicles. This causes the need of hybrid communication, with use case dependent mechanisms for information propagation, to combine the benefits of different network technologies [13]. Moreover, cloud based service frameworks will enable to enrich ADASs with mobility data based services [17].

Hence, we derive various challenges to realize the extension of the vehicular perception range. First of all, use case dependent information needs to be extracted. Whereas we can find several use case listings and classifications in literature, these cover merely some specific types and use different categorization schemes. Therefore, a holistic harmonized view of the respective use cases has to be derived, which we present in Chapter 4.

To realize a long-range view for the vehicular perception, a cellular based collection approach is appropriate. However, data transmission via cellular networks is costly, but then again a maximum detection latency should be guaranteed. As our solution and first key contribution, we present in Chapter 5 our probabilistic information collection approach. By the use of an adaptive probabilistic transmission mechanism, we are able to reduce the amount of data transmissions and still stick to certain quality metrics.

The necessary complement to bring gathered information back to the ADAS in the vehicle, requires also a transmission of a preferable small amount of data. This is achieved in our second key contribution, presented in Chapter 6, by the generation of an eHorizon with selective transmission. Only information, relevant for road segments that will be used, is transmitted.

For the medium perception range, an area of up to 30 seconds remaining driving time till the point of interest, we have identified the additional need of highly dynamic information. Due to the relatively high amount of data and low latency demands, the use of mobile ad-hoc communication is appropriate for information propagation. As our third key contribution, we present in Chapter 7 a mechanism to reduce the channel load for position beacons, by coupling the broadcast to the vehicle dynamics. We use the freed-up bandwidth to forward local knowledge about neighboring vehicles. This allows to more than double the range of vehicular perception, with respect to moving objects.

1.3 THESIS ORGANIZATION

This section gives an overview about the overall thesis organization and the respective structure, as depicted in Figure 2. In Chapter 2 and 3 we present fundamentals and related work and thus, give the technical background to the reader. In Chapter 4 we discuss relevant use cases and derive a schema of three zones of information demands that built up the requirements for respective communication mechanisms. The key contributions are our approaches of efficient data collection and data distribution via cellular networks, that are described in Chapter 5 and 6, and our approach to extend the local perception via vehicular ad-hoc networks, described in Chapter 7. As depicted in Figure 2, the structure of the thesis permits different flows of reading, indicated by arrows. In the following we describe the structure of the chapters:

CHAPTER 1, which is this chapter, gives the motivation with a description of the goals, challenges and respective contributions of this thesis.

CHAPTER 2 introduces fundamentals and definitions, necessary for understanding the thesis at hand. First, we give definitions of terms used in the following of this thesis. Afterwards, we give background about Advanced Driver Assistance Systems (ADAS) and vehicle automation grades, followed by a description of the concept of an electronic horizon. In addition, we give an overview about vehicular communication, digital maps and coordinate systems. Finally we discuss different aspects of position estimation and mechanisms for information distribution, relevant within this thesis.

CHAPTER 3 provides an overview of related work in the specific fields of this thesis. We first discuss aspects of categorization of vehicular use cases, followed

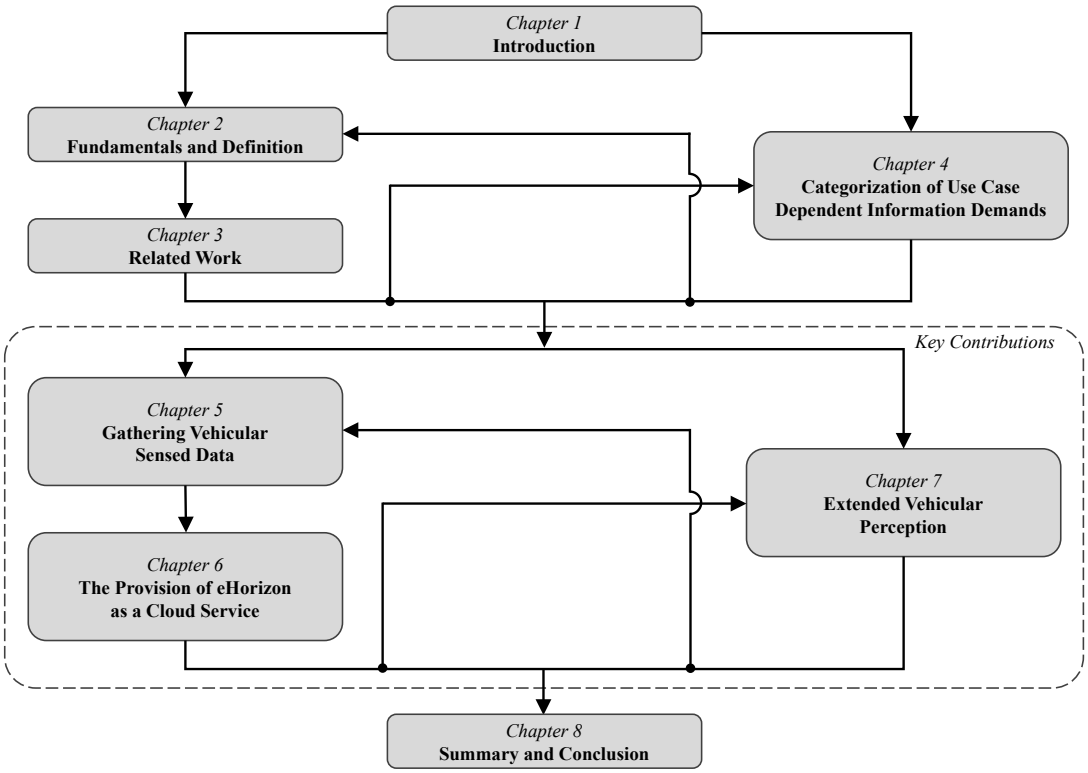

Figure 2: The structure of this thesis with depiction of different flows of reading.

by related work regarding vehicular communication. Afterwards, we give an overview of mobile sensing in vehicular networks, the concept of the electronic Horizon (eHorizon) and remote data for vehicular applications. Following this, we discuss related work with regard to trajectory prediction and vehicle tracking. Finally, we discuss congestion control mechanisms in vehicular ad-hoc networks. Within this chapter, we discuss main differences compared to this thesis and note our scientific contributions.

CHAPTER 4 gives a categorization of use case dependent information demands, with respect to vehicular communication. We have harmonized and restructured use cases described in literature into three zones of information demand. First we discuss aspects of the communication range of Vehicular Ad-hoc NETworks (VANETs) and derive requirements for the three use case zones. Afterwards the three zones of use cases *safety zone*, *awareness zone* and *information zone* are introduced.

CHAPTER 5 describes our approach of gathering vehicular sensed data with probabilistic transmission via cellular networks. We use the redundancy, that potentially many vehicles are passing the same point of interest, e. g., a changed traffic sign. The transmission probability is controlled, individually per property type and geographic cell, to guarantee a certain detection latency and concurrently reduce the amount of transmitted data. After a detailed description of our probabilistic transmission approach, we present our evaluation results, with respect to possible reductions in the amount of transmitted data.

CHAPTER 6 shows in detail our approach of providing route-based information, in terms of the eHorizon as a cloud service. The eHorizon concept allows to efficiently provide driving path related information in contrast to cell based map updates. We present in detail our system concept, followed by a description of the communication architecture with details about the used map matching, most probable path estimation, data serialization structure and the update mechanism. After a definition of data accuracy and granularity, we present our evaluation results with respect to data transmission efficiency.

CHAPTER 7 presents our concept to extend the vehicular perception range in VANETs. We combine trajectory prediction with object tracking by position beacons, to reduce the channel load. The freed-up bandwidth is then used to propagate knowledge about moving objects in the surroundings. For this purpose, we have defined the Cooperative Perception Message (CPM), that brings an extended perception about moving objects to vehicles that receive this message. After a detailed description of our approach, we present our evaluation results with respect to possible channel load reductions for position beacons and to possible perception ranges.

CHAPTER 8 concludes this thesis with a summary of the contents and contributions. Moreover, the chapter outlines potential future work in the specific fields opened up by this work.

2 FUNDAMENTALS AND DEFINITIONS

Within this chapter we provide an overview about fundamentals and definitions that form the technical background of this thesis, as a prerequisite for understanding the following chapters. We describe relevant technical aspects and concepts, and specify basic terms that are used in this thesis. To begin with, we give definitions of fundamental terms that are used within this thesis. Next, we give a definition of Advanced Driver Assistance Systems (ADAS), including a description of vehicle automation grades in Section 2.2. Afterwards, we describe available technologies for vehicular communication in Section 2.4. In Section 2.6 we give an introduction into digital maps and coordinate systems. Finally, we discuss different aspects of position estimation in Section 2.7 and mechanisms for information distribution, relevant within this thesis, in Section 2.8.

2.1 DEFINITION OF FUNDAMENTAL TERMS

Several terms, used within this thesis, are very specific to the area of connected vehicles and ADAS. To avoid misunderstandings, we start with the definition of these fundamental terms.

- **Ego vehicle:** Describes the currently considered vehicle. If vehicles A, B and C are driving close to each other, then vehicles B and C are neighbors in the ego vehicle perspective of vehicle A.
- **ITS station:** An Intelligent Transportation System (ITS) station is a system equipped with information and communication technologies, to support transportation of humans and freight [18]. Purpose is to increase safety, economy and comfort. An ITS station can be communication equipped vehicles, but also stationary Road Side Units (RSUs).
- **Road user:** Basically every user of a road, from vehicles like trucks and cars, over cyclists to pedestrians. Thus, a road user not necessarily has to be a vehicle.
- **Concertina effect:** Causes formation of traffic jams at critical traffic densities. The traffic flow is slowed down by continuous acceleration and deceleration.
- **Variable message sign:** An electronic traffic sign that can change its content. Such systems are typically used to adapt the speed limit.

2.2 ADVANCED DRIVER ASSISTANCE SYSTEMS

Advanced Driver Assistance Systems (ADAS) are electronic systems to improve driving with respect to safety, comfort, economy [19]. Such systems are commonly based on sensor information about system state and environment. The sensing range strongly depends on the used sensor type, as well as the sensor manufacturer. A famous research vehicle, equipped with sensor hardware that is close-to-production,

is the Mercedes-Benz S-Class S 500 *INTELLIGENT DRIVE*, used by Daimler research [20]. The vehicle is equipped with several short range radar sensors with a sensing range of up to 80 m and a view angle of up to 150°, several mid and long range radar sensors, with a sensing range of up to 200 m and a view angle between 18°and 56°, mono cameras with a sensing range of up to 130 m and a view angle of up to 90°and a front directed stereo camera with a sensing range of up to 80 m and a view angle of up to 44°. A further available sensor technology are Light Detection And Ranging (LIDAR) systems with a sensing range of about 200 m. These systems have a very high 3D scanning resolution, but are also very sensitive to attacks [21]. Sensor fusion is used to increase confidence and to make detection plausible.

Famous and already on marked available modern ADAS are blind spot assists, lane keeping assists, adaptive cruise control or traffic sign assists. These systems are completely based on local sensors. However, to further increase possibilities of ADAS and enable higher grades of vehicle automation, the perception range has to be increased. This can be achieved by communication technology and information sharing between vehicles. Examples of possible future ADASs, that need an extended perception view, are systems to support cooperative maneuvers, e.g. cooperative merging at driveways or cooperative overtaking. In a first step, a cooperation partner has to be identified and the cooperation maneuver has to be negotiated. In case of at least one of the involved vehicles is not fully automated, a driver can oversteer and cause an alternative maneuver to be executed. To ensure sufficient reaction time, the maneuver negotiation has to start early enough.

A criteria to classify ADASs is the level of automation. A higher level of automation generally needs a higher perception range. The german federal highway research institute (BASt) defines five levels of vehicle automatization [22]:

- **Driver only:** The human driver executes permanently, i.e. throughout the whole journey, longitudinal guiding (accelerating/decelerating) and transverse control (steering).
- **Driver assisted:** The human driver executes permanently longitudinal or lateral control, the respective other driving task can be an automated system, within certain limits. The driver has to monitor the system permanently and always has to be ready to completely takeover the vehicle guidance. An example is the nowadays available adaptive cruise control.
- **Partly automated:** The system takes over longitudinal and lateral control for a certain period of time or in certain situations. The driver has to monitor the system permanently and always has to be ready to completely takeover the vehicle guidance. An example is the nowadays available traffic jam assistant.
- **Highly automated:** The system takes over longitudinal and lateral control for a certain period of time in specific situations. The driver does not have to monitor the system permanently. If necessary, the driver must take over control with a sufficient time buffer. System boundaries are all recognized by the system. The system is not able to induce the risk minimal state out of every initial situation. Such systems are until now not available, due to legal restrictions.
- **Fully automated:** The system takes fully over longitudinal and lateral control in a defined case of application. The driver has not to monitor the system

permanently. Before finishing the case of application, the system prompts the driver to take over control with a sufficient time buffer. If the driver does not respond, the system induces the minimal risk state. System boundaries are all recognized by the system. The system is always able to induce the minimal risk state. Such systems are until now not available.

A similar categorization is given by SAE J3016 [23]. The only difference in SAE J3016 compared to the definition of *BASt*, is splitting the category *Highly automated* into *Conditional Automation* and *High Automation*, whereas the former has the human driver as fallback and the latter a technical system, i. e., the driver has not to respond appropriately. Both definitions are from a technical point of view, not from a legal.

2.3 THE ELECTRONIC HORIZON

The concept of an eHorizon is to provide digital map information not only for routing purposes, but also to new route based ADASs [24, 6]. Basically the eHorizon provides a long distance view in driving direction. The information source is typically digital map data, that provides information about the road geometry and additional attributes, e. g., the position of traffic signs, curve radii or road gradients. This approach works as virtual sensor, to extend the local sensor perception range [24]. The eHorizon information is provided by a so-called *horizon provider* via a dedicated protocol, named Advanced Driver Assistance Systems Interface Specification (ADASIS), towards other components that implement assistance functions [25, 5]. On the receiving components a so-called *reconstructor* extracts only for the respective component relevant data. The *horizon provider* has access to the local map database and position information, that is used to match the ego vehicle onto the currently used road segment. The eHorizon is constructed as linear path, according to the Most Probable Path (MPP). All attributes are positioned according to their offset as length from the start point. Moreover, a certain depth can be defined, i. e., the number of side paths at junctions. The ADASIS protocol provides four modes of operation. In *mode* 0, only the MPP is transmitted, whereas *mode* 1 also includes information about *stubs*, i. e., intersections. *Mode* 2 further extends the eHorizon by adding the first level of side paths. Finally *mode* 3 transmits the whole eHorizon, according to the given parameters pf maximum length and maximum levels of side paths.

2.4 VEHICULAR COMMUNICATION

Vehicular communication can basically be divided into direct V2V communication, by the use of mobile ad-hoc communication and cellular network based communication. Whereas cellular networks provide a high data rate and network coverage, direct inter-vehicle ad-hoc communication provides much lower communication latencies [15]. However, for both dense traffic is a bottleneck [15].

In the following, we first discuss aspects of the use of cellular communication in vehicles. Afterwards, we explain the state of the art in vehicular ad-hoc communication, based on the IEEE 1609 (WAVE) and European Telecommunications Standards Institute (ETSI) *ITS-G5* standards.

2.4.1 *Cellular Communication*

A possible technology for interconnecting vehicles is the use of cellular networks. An advantage is a high network coverage, at least in urban areas, and a possible high data rate. However, due to the use of commercial licenses, data transmission is always costly. Relevant standards, specified by *3GPP* and ETSI, for the use in the vehicular domain, are Universal Mobile Telecommunications System (UMTS), Long Term Evolution (LTE) and Long Term Evolution Advanced (LTE-A), that are summarized in Table 1 according to [26]. The given transmission rate is the maximum possible transmission rate. Typically, it has to be shared among all subscribers within a single cell and degrades with an increasing distance. Under optimal conditions, LTE allows a round-trip-time below 10ms, i. e., sending a packet to the destination and receiving the acknowledgement. With Multimedia Broadcast Multicast Service (MBMS), respectively evolved Multimedia Broadcast Multicast Service (eMBMS), cellular networks have also a broadcast mechanism, though only for the downlink from base station towards the mobile clients. It supports native IP broadcast and IP multicast within a cell and is designed for the propagation of content like video, weather information or news [18, 26]. Another possible use case, fostered by the low latency, is the propagation of information about moving vehicles.

2.4.2 *Vehicular Ad-Hoc Communication*

Direct inter-vehicle communication, typically named DSRC, is designed to support real time information exchange without the need of any roadside infrastructure [15]. However, DSRC not only supports V2V communication, but also V2I communication, e. g., with RSUs.

In the lower layers, DSRC is based on IEEE 802.11p. In the U.S., as well as in Europe, there is 75 MHz bandwidth in the spectrum at 5.9 GHz allocated for the physical layer of IEEE 802.11p [10, 27]. The PHY and MAC layers of IEEE 802.11p are a modified version of IEEE 802.11a, with Quality of Service (QoS) aspects of IEEE 802.11e, i. e., different packet priorities [14]. To reduce collisions on the wireless channel, Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) is used [14]. According to this, the wireless channel has to be sensed idle for at least the time of one Distributed Coordination Function Interframe Space (DIFS) before trans-

Table 1: Comparison between 802.11p and cellular network standards [26].

Feature	**802.11p**	**UMTS**	**LTE**	**LTE-A**
Channel Width [MHz]	10	5	1.4, 3, 5, 10, 20	Up to 100
Frequency Band [GHz]	5.85-5.92	0.7-2.6	0.7-2.69	0.45-4.99
Transmission Rate [Mb/s]	3-27	2	Up to 300	Up to 1000
Range [km]	1	10	30	30
Mobility up to [km/h]	200	250	350	350
Broad-/Multicast	Native Broadcast	MBMS	eMBMS	eMBMS

mission. If the channel is sensed busy during this DIFS, the sender waits a random backoff time before sensing the channel again. This is to reduce the probability of synchronized transmission attempts after the waiting time [28]. The available bandwidth is split into seven 10 MHz channels, plus a 5 MHz safety margin at the lower frequency end. There is one control channel defined for system control and all safety relevant data and six service channels. The control channel is dedicated to IEEE channel number 180, with a center frequency of 5.9 GHz, in the EU and in the U.S. to IEEE channel number 178, with a center frequency of 5.89 GHz [27, 29]. The usage of these frequencies is not under a commercial license, but restricted to the standards, i. e., it is a "license by rule" [10]. Details about the achievable communication range are considered in Section 4.1.

Whereas IEEE 802.11p specifies the PHY and MAC layers of DSRC, a set of standards on top completes the specification for vehicular communication. In the U.S., IEEE 802.11p, combined with IEEE 1609 protocol family, is named Wireless Access in Vehicular Environments (WAVE) and forms the standard for Vehicular Ad-hoc NETworks (VANET). The IEEE 1609 protocol family describes higher layer functionalities and basically consists of four documents: *IEEE* 1609.1*:* Resource manager specification, *IEEE* 1609.2*:* Security services, *IEEE* 1609.3*:* Network and transport layer services, and *IEEE* 1609.4*:* Support of multichannel operation by enhancement of IEEE 802.11p. In Europe, the counterpart to WAVE is a set of standards and technical specifications defined by ETSI and summarized under the term ETSI *ITS-G5* [27]. The main difference to WAVE is an additional *facilities* layer between network and application layer and that network access is not only focused on ad-hoc networking. *Facilities* are a kind of service, that provide information towards the application layer, i. e., support applications, and independently send out and process messages. Received messages are directly forwarded towards the respective *facility* or application. Example *facilities* are broadcasting of position beacons, in form of the Cooperative Awareness (CA) service, broadcasting warning messages, in form of Decentralized Environment Notification Messages (DENMs) or even communication support, in form of the Decentralized Congestion Control (DCC). Also the management of dynamic information about the environment is specified as *facility* and named Local Dynamic Map (LDM). In ETSI *ITS-G5*, the so called *GeoNetworking* is defined, that enables communication via cellular networks via *IPv6* or IEEE 802.11p based mobile ad-hoc communication with Basic Transport Protocol (BTP) on top. *GeoNetworking* is designed to disseminate safety messages, independent of vehicle density, transport IPv6 packets, and support privacy and security functionality [30]. The following four communication types are supported [31]:

- **Point-to-Point**: Communication is initiated at one node and gets forwarded to another specific node.
- **Point-to-Multipoint**: Communication is initiated at one node and messages get forwarded to multiple neighboring nodes. Maximum propagation in message forwarding is defined by a maximum number of hops.
- **GeoAnycast**: Communication is initiated at one node and messages get forwarded to the first node that is located in the defined geographical area.

- **GeoBroadcast**: Communication is initiated at one node and messages get forwarded to one node that is located in the defined geographical area. The message is rebroadcasted in the defined area.

For addressing nodes, *GeoNetworking* defines an eight byte address, composed of station type (pedestrian, passenger car, bus, etc.), country and link layer address, e. g., the MAC address [32]. For packet forwarding, a *greedy forwarding* and a *contention-based forwarding* is used [32]. In case of *greedy forwarding*, a message is sent by unicast to the known node closest to the destination. Node position knowledge is gained from position beacons. In case of *contention-based forwarding*, the message is broadcasted and forwarded by the receivers after a short delay, calculated based on the distance to the sender. If a receiver listens another node broadcasting the message, that is closer to the destination, then it is discarded. *GeoNetworking* also supports the transmission of *IPv6* packets over the ad-hoc network [33], i. e., a node without Internet connection can send an *IPv6* packet to another node with Internet connection, e. g., a RSU.

The Basic Transport Protocol (BTP) is a connectionless end-to-end transport protocol [34]. Main functionality is multiplexing of messages from different facilities. Equivalent to applications, facilities have a dedicated port number assigned, to forward messages towards it. The BTP has a four bytes header, that either consists of a source and destination port, or just the destination port and an additional port information. Fixed BTP ports for already standardized ETSI *ITS-G5 facilities* are, e. g., 2001 for CAMs or 2002 for DENMs. The major differences between WAVE and ETSI *ITS-G5* are that WAVE supports only one message type, i. e., BSM, to realize position beacons and a broadcast of safety events. Moreover, ETSI *ITS-G5* also supports congestion control and multi hop communication via the previously mentioned *GeoNetworking*. Both is not supported in WAVE.

In the following, we give in addition a brief overview about the most relevant *facilities* and according messages with respect to this thesis.

Cooperative Awareness Message (CAM): The CAM is the European version of the vehicle position beacons, specified in the ETSI *ITS-G5* standard [35]. It is periodically broadcasted with a rate between 1 Hz and 10 Hz and contains the ego-vehicle's position, as well as static and dynamic attributes. CAMs are generated by the CA service *facility*, that also processes incoming CAMs from neighbor vehicles, that can also be forwarded towards other facilities or applications. Information sources are the Vehicle Data Provider (VDP), with access to vehicle status information and the Position and Time management (POTI) entities. Messages are encoded according to the ASN.1 codec specification. The service communicates with the management and security components of the ETSI *ITS-G5* architecture. The CA service manages the generation of CAMs, according to vehicle dynamics and time thresholds. The maximum CAM generation frequency is 10 Hz, that can be down regulated towards a minimum of 1 Hz by the DCC *facility*. In general a new CAM is generated if the ego-vehicle's heading has a minimum change of 4°, the position has a minimum change of 4 m or the velocity has changed more than 0.5 m/s. Moreover, the CAM has optional components, named *low frequency container* and *special vehicle container*, that are attached with a maximum rate of 2 Hz. The U.S. equivalent to the CAM is

named BSM and specified in IEEE 1609 WAVE [36].

Decentralized Environment Notification Message (DENM): A Decentralized Environment Notification Message (DENM) is triggered, if a road hazard or an abnormal traffic condition is detected and contains information about hazard type and position [37]. In ETSI *ITS-G5* four types of DENM are specified: *1) New DENM:* Is generated when the event is detected the first time, *2) Update DENM:* Is used by the same station, that originated the *New DENM*, to update respective information, *3) Cancellation DENM:* Is also used by the same station that originated the *New DENM*, to cancel the validity and *4) Negation DENM:* Is used by any station that sensed that the respective event is no longer valid. The respective *facility* manages message generation, transmission management and management of received DENMs. DENMs can be forwarded by the use of *GeoNetworking* and thus, stay alive until the event is negated.

Local Dynamic Map (LDM): The LDM facility is the central management component for dynamic data from received messages and own data providing *facilities*, e. g., the CA service [38]. It provides data to other *facilities* per query or publish-subscribe. All data includes position, timestamp and time validity. Moreover, a subscription can be associated with a filter that allows to select application relevant information.

Channel Congestion Control: A problem of vehicular ad-hoc communication, or DSRC, is channel congestion, due to the used Carrier Sense Multiple Access (CSMA) mechanism in the MAC layer [16]. In dense traffic, the probability of packet collisions is high, which causes low channel utilization and an increased transmission latency. To reduce this effect, ETSI *ITS-G5* defines the DCC mechanism, that works based on the measured channel load [39]. Moreover, the header of outgoing messages contains information about the ego vehicle's perceived channel load, to inform neighbors. Based on the own measurement and received information, the DCC facility counteracts towards a high channel load. This is basically done by two mechanisms. First of all, the message generation rate is reduced. Therefore, the DCC facility interacts with other services, e. g., the CA service, that reduces the CAM generation rate in five steps down from 10 Hz to 1 Hz. In case the message generation rate is still too high, messages can be dropped. The second adaption is a reduction of the transmit power. Since this is done at all vehicles, the range of interfering vehicles is reduced. The DCC component is organized as cross layer component with subcomponents in the access layer, the networking and transport layer, the facilities layer and the management component, that works across all layers [40]. A more detailed view on channel congestion control mechanisms is given in Section 3.2.

2.5 SECURITY AND PRIVACY CONSIDERATIONS

Security and privacy aspects, with regard to vehicular communication, are in focus of research and standardization since many years. However, security and privacy aspects are not in focus of this thesis. We assume the transferability of research results and already standardized components in this field is given and only highlight some general aspects from standardization. An overview about ETSI *ITS-G5* security services is given in [41]. These components provide authenticity of messages and

privacy to a certain level. The general concept in ETSI *ITS-G5* and WAVE is similar and based on asymmetric cryptography in a Public Key Infrastructure (PKI). Vehicles are equipped with a bunch of pseudonym certificates, that enable the receiver to validate the received message by the use of a certificate from a Trusted Authority (TA). Moreover, vehicles frequently change their sender pseudonym, that is used for identifying a node, to reduce the risk of tracking. This ID change is in particular executed in dense traffic situations, e. g., at crossroads. In the WAVE standard, according security services are specified in *IEEE* 1609.2.

2.6 DIGITAL MAPS AND COORDINATE SYSTEMS

A digital map is an electronic representation of relevant aspects of a particular area. Which aspects are relevant depends on the use case, which results in a scenario of road geometry with attached attributes. Besides several closed source commercial map formats, Open Street Map (OSM)[1] and *OpenDRIVE*[2] are the two major open map platforms, respectively formats. Whereas *OpenDRIVE* is an open map data interchange format, OSM is also a community based world wide map data base.

Open Street Map: The OSM project was initiated in 2004 with the primary objective to establish a global and free editable map. Geographic data is freely accessible and can be maintained or even newly created by everyone. In 2017, there are more than three million registered users, contributing to the open geographic map database with data from their own devices. Moreover, many public authorities and other data providers provide their data to be included. The weekly updated database has a total size of about 50 GB. The data representation is relatively simple and consists of the three elements *node*, *way* and *relation*. A *node* describes a point, consisting of values for latitude and longitude in the *WGS84* coordinate system and some additional describing parameters, named *tags*. A *way* is a polygonal chain, consisting of up to 2000 ordered *nodes*, referenced by their *ID*. *Ways* describe linear features like roads, whereas the type is again given by describing *tag* parameters. A *relation* element consists of an ordered list of *nodes*, *ways* or even other *relations*. An example of a *relation* is an intersection. OSM data is freely available for download, either in *XML* or *Google Protocol Buffers*[3].

OpenDRIVE: *OpenDRIVE* is rather a file format than a database. Its intention is the logical description of road networks, originally for simulation purposes. The format was invented by *Daimler AG* and *VIRES Simulationstechnologie GmbH* in 2005. The latest format specification was released in November 2015 in version 1.4. It can be used free of charge and is vendor independent. The format uses *XML* syntax and describes the road geometry and additionally features the number of lanes or traffic signs. It has a hierarchical tree structure to describe the road geometry and additional features, whereas the geometry elements are the highest hierarchy level. An *OpenDRIVE* file basically consists of the four types of elements *header*, *road*, *controller*, and *junction*. The *header* element consists of the basic file information, like version

1 http://wiki.openstreetmap.org
2 http://www.opendrive.org
3 https://developers.google.com/protocol-buffers/

and area boundary, described by the coordinate range. *Road* elements describe road segments, children elements describe respective features. An example of a feature is a *link* element, that describes linked roads segments, e. g., predecessor or successor, which could be another road segment or a junction. The geometry itself of a *road* element is described in the sub element *plainview*, that is a concatenation of straight lines, spirals, arcs, cubic polynoms and parametric cubic curves. Another sub element is e.g. *lanes*, that describes features like number of lanes. The *controller* element is used to describe states of the traffic signals and the *junction* element describes intersections, i. e., road connections with more than one predecessor or successor.

2.6.1 *Coordinate Systems*

A coordinate system is used to describe the geometry in maps, whereas the used type can cause different processing workload. Typically used coordinate systems to describe road geometry and vehicle positions, are the geographic coordinate system and the cartesian coordinate system.

The surface of the earth is not flat, but it is common consent to approximate the shape as ellipsoid. A coordinate system is based on a pre-calculated ellipsoid, with its origin in the center of the ellipsoid [42]. The mostly used reference system is World Geodetic Datum 1984 (WGS84), globally used for Global Positioning System (GPS). The three dimensional geographic coordinate system describes a position as value pair of *latitude* and *longitude* and is typically used as raw data of positioning systems. In addition, a value for altitude above sea level can be denoted. In order to denote a point on a map, the three dimensional earth model has to be projected into a two dimensional cartesian coordinate system.

Several projection methods have been developed for this purpose, but today commonly used is the cylindrical projection, that uses a virtual cylinder to wrap the earth. All projections have distortion, but the extend is in proportion to the size of the projected area [42]. The rolled out cylinder, in direction south to north, has meridians as equidistant parallel lines, crossed by unequal spaced lines parallel to the equator as geographic grid. Most popular cylindric projections are the *Mercator-* and the *Transverse Mercator Projection*.

Mercator Projection: A widely used cylindrical projection with WGS84 as reference system is the *Mercator projection*. In the normal *Mercator projection*, the cylinder axis is identical with the earth axis and tangential to the equator. With an increasing distance from the equator to the poles, the distortion increases [42]. The radius of a parallel is $R \cdot \cos\phi$ with $R =$ radius of the equator, i. e., parallels have a stretch factor of $(\cos\phi)^{-1}$.

Universal Transverse Mercator Projection: The Universal Transverse Mercator (UTM) projection is not just a single projection [42]. The earth is divided into 60 equidistant zones, each of 6°longitude in width. Numbering starts at 180°to 174°W for zone 1 and is numbered eastwards until zone 60 at 174°to 180°E. In latitude it is divided into 20 bands, each of 8°latitude in hight, between 80°S and 84°N. Lettering starts at 80°S with the letter "C". The last band "X" is 4°larger and thus, ends up at 84°N. For each band an optimal secant transverse Mercator projection is used, which results in

a minimized distortion. Inside each grid zone, represented by identifiers in the form of zone number plus band number, a two dimensional cartesian coordinate system is used. As example, Germany is covered by the four grid zones 32U, 32T, 33U and 33T.

2.6.2 *Line Description*

The geometrical description of road segments in maps is typically a line with attached attributes, like number of *lanes* and *lane width*. This line is described by a polynomial for each road segment. A simple description is a list of vertices, i. e., waypoints, and linear interpolation between these points, i. e., the use of a first degree polynomial. To give a good approximation of the real shape of the road, a high number of vertices is necessary. In contrast, a low number of vertices allows to efficiently describe a road segment, but with a decreasing accuracy. Another possibility to increase accuracy of approximation, is to use higher degree polynomials. A method, popular in the field of Computer Aided Design, is the use of Spline curves, either to connect points or to fit intermediate points. A Spline is a piecewise description of a curve with low degree polynomials. In general, a spline is described by a list of control points and a knot vector. In case of a precise curve description with a lot of vertices, the total data size can be reduced by the use of a Spline, because of the reduced number of necessary data points. An example is presented by Jo and Sunwoo, that uses a B-Spline model to reduce the amount of data in a GPS track [43]. The sensed road geometry is approximated by cubic B-Splines with a set of control points. The authors show a reduction of about 90% in the number of necessary position points to describe the recorded track. The authors show a further reduction of about 75%, by the use of an least-square approximation with a tolerated error of 0.1 m.

A B-Spline curve is defined by a set of control points and an additional knot vector. The knot vector divides the curve into segments. Each curve segment relates to a knot span that is not zero, i. e., the range between two knots. The knot vector $U = (u_0, ..., u_m)$ is typically normalized to the range $[0, 1]$ ($a = 0, b = 1$), with $a =$ the start and $b =$ the end. Thus, the knot span $[u_i, u_{i+1})$ gives the proportion of the length of the curve section to the total length of the curve. The shape of each section of the curve is controlled by a subset of the control points, that are weighted and combined linearly by Equation 1, with P_i: control points (represented as coordinates); $h + 1$: number of control points; p: degree of the B-Spline curve; u: a parameter moving in the knot spans; $N_{i,p}$: basis function for the weight of each control point [44, p. 49].

$$C(u) = \sum_{i=0}^{h} N_{i,p}(u) P_i \tag{1}$$

The order of a B-Spline curve is its degree plus one, i. e., $p + 1$. The value of the weighting $N_{i,p}$ is calculated recursively according to the *Cox-de Boor Algorithm* [45], as given in Equation 2 [44, p. 50]. As mentioned before, $[u_i, u_{i+1})$ is the knot span

between two knots in the knot vector $U = (u_0, ..., u_m)$. u_i is the i-th value in the knot vector U and $u_i < u_{i+1}$ always holds true.

$$N_{i,0}(u) = \begin{cases} 1 & \text{if } u \in [u_i, u_{i+1}) \\ 0 & \text{otherwise} \end{cases} \qquad (2)$$
$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u)$$

The basis function for the weight of a control point of degree 0, i. e., $p = 0$, has the value 1, if it is on the respective knot span, decided by the parameter u. If it is not related to the respective knot span, the value is 0. The algorithm iterates the degree from 0 to p, which results in only some of the control points affecting the respective section of the B-Spline curve. This means that a control point is locally tuning a B-Spline curve, but does not have an effect on the curve globally.

The knot vector specifies the connection points of two curve segments. The number of knots m in the knot vector is defined by the degree of the B-Spline p and the number of control points h, as given in Equation 3.

$$m = h + p + 1 \qquad (3)$$

To ensure the start of a B-Spline curve in its first control point, the first $p + 1$ knots must have the value 0 and to ensure the end of a B-Spline curve in its last control point, the last $p + 1$ knots must have the value 1 [44, p. 66]. The rest of the knots can be evenly spread, which results in Equation 4.

$$\begin{cases} u_i = 0 & i \in [0, p] \\ u_i = \frac{i-p}{h-p+1} & i \in [p+1, h] \\ u_i = 1 & i \in [h+1, h+p+1] \end{cases} \qquad (4)$$

Curve approximation: The general idea of using a B-Spline to model the path of a road segment, is to reduce amount of data for road path description. We assume the initial path description as polygonal chain, i. e., an ordered list of position points. To reduce the amount of data, the number of points $N_D = n + 1$ (n plus one, because the index of n starts at zero) to describe the path has to be reduced . As one possibility in the resulting curve fitting problem, a B-Spline of degree p, defined by $h + 1$ control points $(P_0, ..., P_h)$, is used to approximate the path described by $n + 1$ data points $(D_0, ..., D_n)$, with the method of least squares. The B-Spline curve starts in $P_0 = D_0$ and ends up in $(P_h = D_n)$, with a possible curve fitting approximation as follows [44, 46].

The B-Spline is described by Equation 1. A parameter vector $(t_0, ..., t_n)$ can be generated as equally distributed values $t_i = i/m, i \in [0, m]$. The approximation with the method of least squares is given in Equation 5. D_k is the k_{th} data point. $C(t_k)$ is the value of the B-Spline for the k-th element of the parameter vector $(t_0, ..., t_n)$.

$$f(P_1, ..., P_{h-1}) = \sum_{k=1}^{h-1} |D_k - C(t_k)|^2 \qquad (5)$$

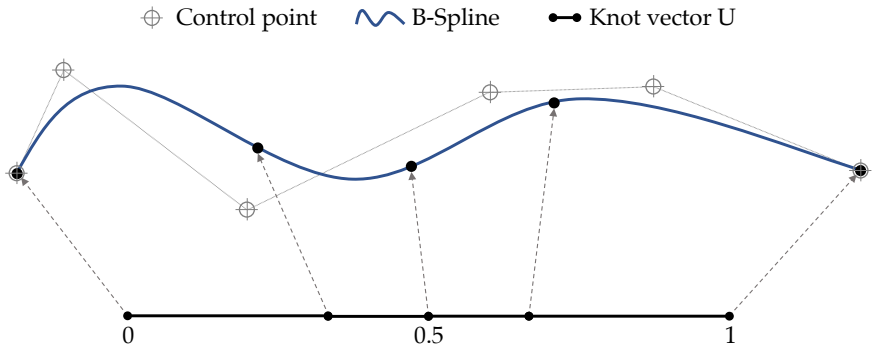

Figure 3: Example illustration of a B-Spline curve and the respective knot vector U. The relative length of the knot spans relates to the length of the respective curve segments.

Next, the control points can be calculated according to Equation 6. Note that $P_0 = D_0$ and $P_h = D_n$, i. e., the total set of control points is $P = (D_0, P_1, \ldots\ \ldots, P_{h-1}, D_n)$.

$$P = (N^T N)^{-1} Q \qquad Q_k = D_k - N_{0,p}(t_k) D_0 - N_{h,p}(t_k) D_n \tag{6}$$

$$P = \begin{bmatrix} P_1 \\ P_2 \\ \vdots \\ P_{h-1} \end{bmatrix} \quad Q = \begin{bmatrix} \sum_{k=1}^{n-1} N_{1,p}(t_k) Q_k \\ \sum_{k=1}^{n-1} N_{2,p}(t_k) Q_k \\ \vdots \\ \sum_{k=1}^{n-1} N_{h-1,p}(t_k) Q_k \end{bmatrix} \quad N = \begin{bmatrix} N_{1,p}(t_1) & N_{2,p}(t_1) & \cdots & N_{h-1,p}(t_1) \\ N_{1,p}(t_2) & N_{2,p}(t_2) & \cdots & N_{h-1,p}(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ N_{1,p}(t_{n-1}) & N_{2,p}(t_{n-1}) & \cdots & N_{h-1,p}(t_{n-1}) \end{bmatrix}$$

In our context, we use the B-Spline curve to approximate the road geometry, defined by a polygonal chain of position points. The B-Spline gives the relative position on the road segment, with respect to the start point. The offset is normalized into the range $[0, 1]$ of the knot vector. The relation of moving the parameter u from 0 to 1 is equal to the distance in the movement from the start point of the road segment to the end. Thus, u is calculated by the offset of the position, divided by the total length of the road segment. An example of a B-Spline curve and the respective knot vector U is illustrated in Figure 3.

2.7 POSITION ESTIMATION

Position estimation with respect to automotive scenarios, consists of different perspectives. First of all, it relates to the process on estimating the relative position on a road segment and in particular determining the correct road segment. This process is typically named *Map Matching*. The problems are errors in Global Navigation Satellite Systems (GNSS) like GPS, especially in urban areas. An extreme case could be driving in a tunnel or parking garage, where GNSS signals are not available. To compensate these measurement errors or loss in the positioning signal, the local positioning component can use a Kalman filter to integrate local available sensor data like the distance counter of the odometer or the vehicle heading. This technique of mixed source position determination is named *dead reckoning*.

With regard to the mentioned *Map Matching* problem, besides several other approaches, Quddus et al. introduced an advanced matching approach with low processing demands [47]. The authors use the recent position history, combined with

Kalman filtered *dead reckoning* and a map database, to reliably match the vehicle position on a map. The algorithm is very efficient and accurate, in particular in complex situations like intersections. The matching is based on a weighting scheme for possible links, i. e., connected road segments at an intersection. In a first step, all possible links are selected from the database. The weighting is calculated by the difference of the Kalman filtered heading of the positioning component to the respective angle of the road segment, according to the map data. According to the authors, similarity in orientation to a road segment is more important than proximity.

The previously mentioned Kalman filter is a mathematical method to estimate the next state of a system, that is corrected with a measurement [48]. Also several other filter models are available, like particle filters that are good for nonlinear systems [49]. However, in the application of position estimation in the vehicular domain, i. e., *dead reckoning*, the extended Kalman filter is commonly used [47, 50]. The Kalman filter is an optimal estimator, if the measurement noise is Gaussian and the prediction process is linear, i. e., it minimizes the mean square error of the estimated state. In case of the process to be estimated can not be described by a linear prediction function, the extended Kalman filter is used, that linearizes the function in its current state. The basic principle is based on a cycle of prediction, measurement and correction. In the example of position estimation of a vehicle, the Kalman filter is used for *dead reckoning*. The vehicle position is described as state vector, e. g., with position, velocity, acceleration and heading. In the prediction step, this state information is used to predict the next state, i. e., the new position. This prediction is based on a motion model, e. g., a simple constant velocity or constant acceleration model. Due to the uncertainty, modeled as Gaussian distribution, the variance of the predicted state increases, i. e., it is the sum of the variance of the previous state and the variance of the prediction. In the next step, the position is measured by GNSS signals, e. g., GPS. The measurement state is also Gaussian distributed, with a certain variance. The measured state is then multiplied with the predicted state and the *Kalman Gain*, that controls how fast the filter converges towards the measured values. The filter periodically overwrites the mean value and the variance of the result. On the measurement step, the certainty increases and on prediction it decreases. Overall the certainty increases, as long as the measured values do not deviate too much from the predicted value.

This concept is the basic inspiration for our simplified position prediction mechanism used in Chapter 7. However, the prediction we use is not a Kalman filer, because we give 100% certainty to the measurement and just overwrite the predicted value.

2.8 INFORMATION DISTRIBUTION

In context of this work, information distribution towards vehicles can either be done via direct V2V communication or provided by a backend. Whereas V2V communication in general is considered in more detail in Section 2.4, we consider in the following some aspects of providing information by a backend. If a mobile node is interested in constant information updates, provided by a backend, basically two approaches can be used, namely a push or a pull based approach [51]. In case of a pull approach, the potential data traffic is significantly higher, compared to a push based approach, if no new data is available [52]. An appropriate technology for realizing

a pull based approach is the use of web services [51, 53]. However, web services typically make use of Hypertext Transfer Protocol (HTTP) and JavaScript Object Notation (JSON) or Extensible Markup Language (XML) for data serialization, which causes a significant protocol overhead [53]. An alternative is a push based approach. Here, the mobile client subscribes to a specific interest and the backend immediately publishes new information towards the mobile client as push notification [51]. In the publish-subscribe pattern, the publisher sends a new message towards a message broker and is itself unaware of the receivers [54]. The message broker publishes, i. e., sends, these messages towards the mobile clients, according to filters previously set. A subscriber specifies its request as a subscription to a set of filter criteria. Such filters can be *topic* based or *content* based [54]. In case of a *topic* based approach, a sender publishes a message towards a specific topic. A *topic* can be described as logical channel. All clients with a subscription to this respective topic will receive this message. In case of a *content* based approach, the principle is the same, but the respective filter is more complex. The subscriber defines a filter as constraints. If all constraints apply on a newly published message, the respective subscribers will receive this message.

A message broker is a message oriented middleware, that works as intermediary, which routes messages from the publishers to the subscribers [54]. Subscribers register their subscriptions at this broker, that typically also performs the message filtering. Operations of the message broker are to store and forward messages, including prioritization of messages and to queue messages. Also realizations without such a middleware are possible, but the synchronization overhead of publishers and subscribes increases in case of non static systems, i. e., a system of mobile nodes.

For realization, a communication protocol is necessary, to enable the message exchange between all entities. In the lower layers, message broker based publish-subscribe systems typically make use of *TCP/IP* based communication, but this is not mandatory. The subscribers maintain an open *TCP* socket connection to the message broker, that is used for message delivery. Thus, the client is responsible to be available for message delivery. On top, an application layer protocol is necessary to manage connection establishment, message delivery, prioritization, authentication, QoS and in particular to manage subscriptions.

A very efficient publish-subscribe protocol is Message Queuing Telemetry Transport (MQTT)[4]. It is designed for mobile applications and IoT, has a very low header overhead with a fixed header size of only two bytes and supports Transport Layer Security (TLS) based encrypted connections. The protocol was introduced by *IBM* and *Cirrus Link Solutions*, standardized by *OASIS*[5] since 2014 as communication protocol for IoT and *ISO* standard since June 2016 (c.f. *ISO/IEC 20922:2016*). Also several other publish-subscribe protocols like *AMQP* or *XMPP* are available, but these have typically a much larger communication footprint. Thus, we have used the MQTT protocol in our prototypes, in combination with the lightweight open source message broker *Mosquitto*[6].

A special case of publish-subscribe supported information dissemination are *geocasts*, where messages are delivered according a geographic location [55]. All clients inside the respective geographic location, that can be any definable area like e. g. an area defined by a closed polygonal chain, an area of a circle or a whole city area, will

4 http://mqtt.org
5 https://www.oasis-open.org
6 https://mosquitto.org

receive the message. Here, an overlay for message delivery, like a middleware based publish-subscribe, is necessary, because the concept of the Internet is not designed for mobility. Node addressing via *IP* addresses combines the locator and node identifier within a single address. As a result, the *IP* address does not allow to infer the position of a mobile node. Thus, in case of *geocasts*, the publish-subscribe middleware maps subscribers to geographic regions. As a consequence, mobile nodes have to propagate their positions towards the middleware. In case of using a publish-subscribe message broker, this can be done by subscribing towards the respective geographic region, the mobile client is currently in. This brings two benefits: First, a position update is only necessary on changing geographic regions, which reduces the according data transmission overhead. Secondly, the middleware does not know the exact position of mobile clients inside a geographic cell, which brings a certain level of privacy.

3 RELATED WORK

Within this third chapter we discuss related work regarding our contributions, presented in Chapters 4 to 7. We start with different attempts to categorize vehicular use cases, with respect to information exchange in Section 3.1. Following this, we look into vehicular communication with respect to channel congestion and object tracking in Section 3.2. In Section 3.3 we analyze mobile sensing approaches, used within vehicular scenarios. Afterwards, we discuss related work about the electronic horizon in Section 3.4. Finally, we conclude this chapter in Section 3.5 with a summarizing discussion and outline the problem statement that is addressed within this thesis.

3.1 CATEGORIZATION OF VEHICULAR USE CASES

Typically three safety relevant use cases are often mentioned with regard to communication robustness [56]. These are cooperative collision warning, electronic emergency brake light and the slow or stopped vehicle alert. Sepulcre et al. consider application requirements of a lane change assistant, with respect to the communication needs [57]. They assume a minimum safety distance D_{ω} and define that at least one of two affected vehicles has to broadcast at least one position beacon within the communication range, before the relative distance undercuts D_{ω}. Thus, without a preceding cooperative maneuver agreement, such applications also belong to safety relevant use cases.

Zheng et al. propose the use of heterogenous networks, in particular the combination of DSRC and cellular networks [15]. They propose a respective communication framework, based on a use case study. Within this study they basically differentiate between two major categories, *safety related* and *non-safety related*. For safety relevant use cases they stick to periodic messages with a maximum latency of 100 ms. Non-safety relevant use cases are split into two subcategories *traffic management* and *infotainment*. For both, they still stick to a minimum frequency for periodic messages of 1 Hz. The authors highlight that both, cellular networks and DSRC, have drawbacks in information exchange, with regard to different use cases and propose the use of DSRC for direct V2V communication and the use of cellular networks, in particular LTE, for V2I communication.

Karagiannis et al. present a survey about vehicular networking and also analyze applications and requirements [58]. They use the three application categories *active road safety*, *traffic efficiency and management* and *infotainment*. Although they stick to a fixed transmission rate of 10 Hz for *active road safety* related messages, they mention a rate of 1 Hz for the other use cases. A simplification towards only two categories is used by Armaghan et al. [59]. The authors only name the categorizations *comfort* and *safety*, whereas traffic efficiency also belongs to *comfort*. A similar categorization is *cooperative vehicular safety* and *convenience & efficiency* applications [60].

Bai et al. classify automotive applications with regard to wireless networking aspects [61]. The authors consider application characteristics and networking attributes.

The three basic named application categories are *safety applications*, like post crash notification, cooperative collision warning or road hazard condition warning, *convenience applications*, like free flow tolling or parking availability notification, and *commercial applications*, like map downloads or video streaming. The authors consider different information dissemination strategies and propose an appropriate network protocol stack for each class of application.

Toor et al. differentiate between *safety applications* and *user applications* [62]. With regard to inter connecting vehicles, the authors consider infrastructure-based networks as well as ad-hoc networks. Within *safety applications*, the major use cases relate to accident prevention, intersection assistance and road congestion. The focus of *user applications* relates to Internet connectivity [63] and peer-to-peer connectivity [64] for multimedia sharing between vehicles. The authors also consider respective routing strategies.

Uhlemann describes connected vehicles from the application perspective, with a focus on autonomous driving [65]. The era of self-driving vehicles is coming and brings new demands to vehicular connectivity, in particular for *infotainment*. This change brings also changes in cellular networks, that are already applied to big field tests like the *digital A9 motorway testbed*. Cell tower base stations are getting equipped with server technology to become cloudlets, that are able to provide high speed inter-vehicle communication. This will serve as supplementary technology to V2X and provides the means to provide long-range perception information. Mentioned use cases relate to driving efficiency and comfort, e. g., traffic light phase information.

Wachenfeld et al. present a comprehensive consideration of use cases, with respect to autonomous driving [8]. The authors basically distinguish the three categories of driving tasks *navigation*, *path tracking* and *control*. They describe *navigation* as the respective route selection, *path tracking* as deriving command variables for guidance and *control* as corrective actions for stabilization. Within all three types, certain parameter inputs can manipulate the respective task execution. Selected characteristics to describe the use cases are *type of occupant* (e.g. humans or cargo), *maximum permitted gross weight*, *maximum deployment velocity*, *scenery* (e.g. road or parking lot), *dynamic elements* (types of other objects to interact with, e.g., vehicles, cyclists or pedestrians), *information flow* (which information is exchanged and how) and *handover to a specified availability or extension concept* (e.g. tele-operated driving or human driver as backup). As example, the *availability concept* influences the necessary range of perception, due to handover time.

The LDM is the ETSI standard for an in-vehicle database facility to manage information about the vehicle surroundings [38]. Information is not considered related to the object they belong to, but to the level of dynamics of the single property. The used four categories of information are *permanent static* (e.g. road topography or statutory speed limits), *transient static* (e.g traffic signs), *transient dynamic* (e.g. road works or temporary speed limits) and *highly dynamic* (e.g. current speed, position and direction of other vehicles).

Weiß gives an overview about research on vehicular communication and describes perspectives of safety applications [66]. He highlights the possibility to extend the perception view of local sensors, by the use of communication and names communication as a *sensor*. A special aspect of safety is the time a driver has to adapt to a situation, i. e., a long-range perception can prevent safety critical situations. As the main application categories the author names *safety*, *traffic efficiency* and *(commercial)*

services. Moreover, the author also considers different information flows, with respect to communicating entities.

Al-Sultan et al. present a comprehensive survey about VANET technology and respective applications [67]. The authors basically differentiate between *comfort & entertainment applications* and *safety applications*. All non-safety applications, to improve driving comfort and traffic efficiency, are summarized in *comfort & entertainment applications*. However, the focus is on *safety applications*, which are subcategorized into *intersection collision avoidance*, *public safety*, *sign extension*, *vehicle diagnostics & maintenance* and *information from other vehicles*. Using this categorization approach, safety related messages are not directly coupled to strict time constraints. *Vehicle diagnostics & maintenance* includes *just-in-time repair notification* as an example for a safety application.

The ETSI documents about vehicular communications describe a large set of possible applications [68, 69, 70]. The respective use cases are considered from three perspectives: First, from a general perspective as *basic set of applications*, second, as *road hazard warning* and finally, in the perspective of *longitudinal collision risk warning*. Use cases are grouped into *active road safety*, with the subgroups *cooperative awareness* and *road hazard warning*, *cooperative traffic efficiency*, with the subgroups *speed management* and *cooperative navigation*, *location based services* and *global Internet services*, with the subgroups *communities services* and *ITS station lifecycle management*. Also latency demands are considered, with respect to information processing, from data acquisition from sensors, till the required action. These demands are categorized into the two major classes *driving assistance* and *direct control*. The direct control addresses automatic execution, as well as pre crash and post crash actions, e. g., an emergency call. The *driving assistance* is divided into the three subclasses *info*, *awareness* and *warning*. The *awareness* class basically consists of road hazard warning use cases, whereas *warning* basically consists of collision risk warning use cases.

Möbus et al. present vehicular applications, with a need of long-range perception, in the context of the eHorizon [5]. They describe how a set of *safety* and *comfort* applications can significantly benefit from prescient knowledge road network related information. Also Durekovic and Smith focus on the eHorizon and categorize ADASs into *non-map ADAS*, *map-enhanced ADAS* and *map-enabled ADAS* [71].

Based on the obtained knowledge about vehicular use case demands, we have derived a scheme of three zones of information demands. We differentiate the zones by the tolerated latency, between the occurrence of an information and the point in time, it has to be processed at the receiver side. These three zones are named *safety zone*, *awareness zone* and *information zone*, that are described in detail in Chapter 4 and a description of the single use cases belonging to each zone can be found in Section A.1.

3.2 VEHICULAR COMMUNICATION AND CONGESTION CONTROL

As introduced in Section 2.4.2, inter vehicular communication is based on IEEE 802.11p mobile ad-hoc communication in the 5.9 GHz band. In general, vehicular networks can be considered from different perspectives, like routing protocols, delay tolerant networks or channel load. With regard to Chapter 7, we have the channel load in scope of this work, because of the high amount of information, that has to be

shared for a long-range vehicular perception. Related work in this field can be split into congestion control in general and an optimized position beacon control.

In IEEE 802.11p, the MAC layer uses CSMA/CA, i. e., it follows a listen-before-talk-principle. If the channel is sensed free, the sender waits a random backoff time before sending. In high vehicle densities, the wireless medium quickly becomes congested with a high probability of packet collisions [14, 72, 73]. Although channel congestion has been studied in the field of mobile ad-hoc networks (MANET), settings in vehicular communication completely differ [74]. Most important applications relay on tracking neighboring vehicles, achieved by periodic awareness messages, i. e., CAMs or BSMs [73].

An example for congestion control, with regard to message forwarding, is given in [75]. The authors introduce a retransmission probability to adjust the channel load. By decreasing the retransmission probability, the overall channel load is reduced. Similar to this, also other approaches often focus on retransmission, but in case of vehicular communication, the periodic broadcast of position beacons, combined with high node mobility is a different case.

Since packet collisions in CSMA/CA is a well known problem, the European ETSI *ITS-G5* standard already defines a Decentralized Congestion Control (DCC) mechanism [39, 40]. In general, the network performance is affected by the communication range and the transmission rate of CA messages [73]. The value of the measured channel load is added to the message header of broadcasted CA messages. In case of a high channel load, the CAM generation rate is controlled down in five classes from 10 Hz to 1 Hz. In addition, also the transmit power is reduced in case of a high channel load [40].

A second component of the ETSI *ITS-G5* Decentralized Congestion Control (DCC) mechanism is a *gatekeeper* between the access and networking layer [39, 76]. It adds an additional delay to packets in the output buffer in case of channel congestion. Since this happens independent of the packet content, the overall performance of applications may suffer. However, since the channel load is still close to its limits in dense traffic situations, several other approaches have been proposed in research.

Many approaches focus on solving this issue as a networking and communication problem, like the distributed power control approach, presented by Torrent-Moreno et al. [77]. The authors propose to reduce the transmission power in case of a high channel load. The general goal is to achieve fairness, by controlling towards a target channel utilization rate. In case this target rate is exceeded, nodes decrease the transmit power in predefined steps and vice versa, in case of a channel utilization lower than the threshold. In addition, the proposed solution uses a contention mechanism to prioritize emergency messages. Baldessari et al. propose to achieve fairness by a conjoint control of power and transmission rate adaptation [78]. The authors suggest to decrease the transmit power in case of an increase of the transmit rate and vice versa.

Tielert et al. introduced the *Periodically Updated Load Sensitive Adaptive Rate (PULSAR)* mechanism [79]. The mechanism adapts the position beacon transmission rate, based on the measured Channel Busy Ratio (CBR), that is also broadcasted towards neighboring nodes. Received CBR values are also forwarded and the transmission rate calculation is based on the maximum congestion state in the 2-hop distance. The authors use a first-order low-pass filter to smooth the CBR calculation, that is performed in 250 ms intervals. The rate adaption is performed as additive increase

multiplicative decrease, based on a binary decision if a CBR threshold is violated. All vehicles, that are contributing to channel congestion, are also participating in rate adaptation. However, even if this approach achieves a good global fairness, it still shows limit cycle behavior.

In contrast to this, in the *Linear Integrated Message Rate Control (LIMERIC)* algorithm the CAM broadcast rate converges towards a predefined threshold, but remains below it [80]. *LIMERIC* uses linear full precision control inputs to avoid the limit cycle behavior. The approach ensures that all nodes, within a congestion area, converge against the same message rate. Instead of a binary decision, if a channel is congested or not, in *LIMERIC* each node measures the fraction of network capacity at each moment and compares it against the message rate goal. The own allowed broadcast rate is linearly adjusted to a transmission rate, calculated with the difference of the desired rate goal and the measured value. As a result, the wireless channel is shared fair at a specified CBR threshold. It has shown, that the adaptive *LIMERIC* approach outperforms the reactive ETSI standardized DCC approach [81]. Rostami et al. give a detailed performance comparison of the reactive DCC approach and the adaptive *LIMERIC* approach [82]. The authors show, that the reactive DCC approach can have strong oscillations in CBR. The authors identify a relatively synchronized CBR measurement, with a too limited range of message rates, as main cause for instability. The ETSI *ITS-G5* DCC mechanism provides exactly five CAM transmission rates, between 10 Hz and 1 Hz. These are respectively selected if a CBR below 30%, in 10% steps between 30% and 60%, or above 60%, is measured. The CBR class is selected, according to the maximum value measured, within the last $T_{down} = 5s$ [82]. To solve this issue, the authors propose to use a linear scale for the transmission rate adaptation and more important, to have an asynchronous CBR measurement across all vehicles.

Cheng et al. have analyzed the mixed mode of vehicles, equipped with the ETSI *ITS-G5* DCC mechanism and *LIMERIC* [83]. It has shown that the overall performance increases, if *LIMERIC* vehicles are inserted. This is mainly caused by the asynchronous CBR measurement in *LIMERIC*, that also allows the ETSI *ITS-G5* DCC to converge better. Thus, also a mixed mode deployment might be possible, e. g., by a later update of the used DCC mechanism.

A weighted version of *LIMERIC* is introduced by Bansal and Kenney in [84]. The idea is to converge globally against a fixed CBR rate, but locally against a weighted value of this rate. This allows to provide more resources to some dedicated nodes, e. g., nodes that constitute a higher safety risk. A context based approach is presented by Sepulcre et al., that also reduces transmit power and the CAM broadcast rate [57]. They consider the application of a lane change assistant and increase the CAM broadcast rate only, if a lane change is reasonable. Received CAMs are evaluated and in case the context does not allow a lane change, i. e., minimum vehicle distances are not observed, the broadcast rate is reduced to keep the channel load low.

Egea-Lopez and Pavon-Mariño address the transmission rate control as an optimization problem in the *Adaptive Beaconing Rate for Intervehicular Communications (FABRIC)* approach [85]. The position beacon rate is modeled as a network utility maximization problem. Vehicles exchange information about their current congestion state. The own congestion state and the received values from the neighbor vehicles are used in a local utility function, that has to be maximized. The approach achieves fair beaconing rates and controls fast out of channel congestion. A simi-

lar approach is proposed by Jose et al., that also regulates transmission power, in addition to the broadcast rate [86].

Piao et al. propose a much simpler approach, that only depends on the estimated amount of neighbor vehicles and has no need for CBR measurements [87]. They start with a broadcast rate of 10 Hz and reduce this rate, if the amount of neighbor vehicles increases. In case the rate reduction is strong enough, message delivery characteristics are better than other DCC approaches. However, a reduction of the broadcast rate would require a good trajectory prediction, to prevent a suffering in accuracy.

Aygun et al. propose to adjust the broadcast rate and transmission power, according to the environment and context [88]. The mechanism proactively considers the mutual effect of rate and power adjustment to adapt more efficiently. The adaptation is based on CBR, dependent on the current application requirements and environment, e. g., intersections or highways. The used rate control algorithm is based on *LIMERIC*, but used in combination with a transmit power control mechanism. In addition to the measured CBR, also the current transmit power value is added to the message header of broadcast messages. The transmit power is adjusted to the current application requirements, as well as to the current channel path loss, that is estimated based on received transmit power values. The algorithm is able to increase the cooperative awareness by about 20%, while keeping a given channel utilization.

A central controlled approach to manage congestion control at intersections is presented by Taherkhani and Pierre [89]. A RSU at the intersection observes the wireless channel to measure and control channel congestion. Vehicles, contributing to the congestion, are clustered into four groups by the use of a *k-means* machine learning mechanism. Each cluster is provided with individual communication parameters, including transmission rate, transmission power, contention window size, i. e., the maximum backoff time, and Arbitration Inter-Frame Spacing (AIFS), i. e., the minimum time, the channel has to be sensed clear before transmission. The approach basically addresses vehicles stopped at a red light at an intersection.

Another possibility to reduce the channel load is to reduce the message broadcast rate, based on an estimated tracking error of neighboring vehicles, i. e., only broadcast, if a neighbor would not be able to estimate the ego vehicle position within a certain threshold [90, 72]. This concept has been initially introduced by Rezaei et al. in [90]. A position estimator, based on an extended Kalman filter, is used to each known neighbor vehicle to continuously estimate its position, based on the history of received position information. The same estimator is also used at the sender vehicle, named *self estimator*, with only the broadcasted position information as an input. The used extended Kalman filter is based on a constant-velocity-constant-heading model. A scheduler observes the relative error in position between the output of the *self estimator* and the assumed real position of the ego vehicle, i. e., the suspected tracking error. If the threshold is exceeded, a new position beacon is broadcasted, that is also consisting of current heading and velocity. The approach has an adaptive message rate with an average broadcast rate of about 2 Hz. The authors add a message repetition mechanism in [72] and show that the application level performance cannot be measured by packet loss. Thus, they propose to use the tracking error in presence of packet loss as performance metrics.

Armaghan et al. use a *Kalman* filter to predict the longitudinal position of the ego vehicle [59]. Then, a position beacon not only consists of the current position, but

also of a set of predicted future positions. A new beacon is broadcasted at the end of the predicted positions or when the suspected tracking error exceeds a threshold.

Segata et al. propose the adaptive vehicle state dependent protocol *Jerk Beaconing*, in the application field of platooning [91]. The approach makes use of a constant-acceleration-constant-heading-model and has a minimum beacon rate to ensure awareness. It tries to minimize the beacon rate and only broadcasts a new beacon in case the acceleration changes above a threshold. However, the approach incorporates acknowledgements, that is only possible because of the relatively fixed amount of participating vehicles.

Other approaches combine the networking and communication problem with an estimation process, like the *InterVehicle Transmit Rate and Power Control (IVTR-PC)* algorithm presented by Huang et al. [56]. In a first step, Huang et al. extend the position estimation error approach with a message rate control mechanism to reduce the channel load [92]. The trigger check rate is set to an increased rate of 20 Hz and the CBR is measured continuously. At each time step, the transmission probability for a new position beacon is calculated as function of the suspected tracking error and the CBR. The transmission probability converges to one, in case of a high tracking error or a high CBR. The algorithm shows a good tracking accuracy, while simultaneously being robust to channel congestion [93]. Huang et al. further extend this approach by additional transmit power control in [56]. In general, the algorithm increases the broadcast rate in case of an increasing estimation error and throttle the broadcast rate during channel congestion. Messages are only broadcasted in case the suspected tracking error exceeds a threshold e_{th}. When e_{th} is exceeded, a message is broadcasted with an exponentially increasing probability, according to an increasing tracking error. Moreover, the CBR is measured as an average over a 1 s window and in case a lower threshold is exceeded, the transmit power is linearly reduced. At a certain upper CBR threshold, the transmit power is set to a predefined minimum. The used kinematic model for vehicle tracking is here also a constant-velocity-constant-heading-model.

The *Error Model Based Adaptive Rate Control (EMBARC)* approach combines the mechanisms of previously mentioned *LIMERIC* and *(PULSAR)* [94]. Moreover, the *EMBARC* approach also uses the position estimation error approach, introduced in *(IVTR-PC)* [93] . In addition, the Packet Error Rate (PER) is incorporated in calculation of the suspected tracking error, that is the sum of each step's product of the calculated tracking error ε_n and one minus the PER p_n. In case of an increased suspected tracking error, i. e., increased vehicle dynamics, the broadcast rate of position beacons is increased. Thus, the *EMBARC* approach adjusts the transmission rate, according to channel load and vehicle dynamics, to maintain the CBR at or below a specified threshold. Simulation has shown that *EMBARC*, as combination of channel load and tracking based approaches, outperforms *IVTR-PC* and *LIMERIC*, with respect to tracking accuracy. The *EMBARC* approach is a strong candidate to become part of the U.S WAVE standard [87].

Sepulcre et al. introduced *INTEgRatioN of congestion and awareness control (INTERN)*, an adaptive approach that integrates *LIMERIC* and *PULSAR* in [95]. The minimum broadcast rate is controlled according to the individual minimum application requirements of each vehicle, whereas overall channel load is maintained at or below a target CBR threshold. In case of a high CBR, the broadcast rate is reduced until the minimum application requirement, otherwise the mechanism tries to fully uti-

lize the channel. To achieve global fairness, the 2-hop congestion state propagation of *PULSAR* is used.

3.3 MOBILE SENSING

Mobile sensing is an important domain within our contributions in Chapter 5. We have discussed this topic in our previous work [96] and revise in the following. Within the field of mobile sensing, smartphones are often discussed as sensing devices, because of their large amount of sensors and mobility [97]. A related domain is Mobile Crowd Sourcing (MCS), where a huge amount of users with mobile devices are the basis to build large scale sensing applications [98]. To distinguish between different sensing models, we categorize them by three criteria: The amount of subjects participating, the degree of human participation, and the treatment of collected sensor data. The amount of participants allows to subdivide sensing models into the three groups personal, group, and community sensing. However, in our work and as well as in MCS only community sensing is of interest. With regard to user interaction, one can distinguish between participatory sensing and opportunistic sensing. The participatory sensing model applies user interaction to collect data and focuses on applications to help users to share, search, interpret, and verify information [99, 97]. Due to the superior intelligence of the user, complex data can be collected. However, the quality also depends on the users. A major problem is that the user must be actively involved in the collection process. Hence, the task to encourage users to take part is quite challenging [100]. This may, for example, be addressed by requiring active participation by the user, before allowing access to the collected data. However, in vehicular scenarios the participatory sensing model is not applicable, due to the active involvement of the driver.

Sensing models are named opportunistic, if no active user participation is required, that is beneficial in vehicular scenarios. This model is commonly used in vehicular monitoring applications, i. e., vehicles transmit data about their environment [101]. According to Shin et al. , opportunistic sensing has to cope three basic challenges [101]: To protect user privacy, to ensure data quality and integrity, and to consider an efficient data transmission, since typically all sensed information is directly transmitted. The sensing model can be further categorized into probabilistic and deterministic sensing, depending on the treatment of gathered information. In deterministic sensing, all sensed data is sent. For this approach, complex maintenance or processing can be neglected on the sender side, but network traffic might become a problem.

In contrast to this, information will only be transmitted with a certain probability in probabilistic sensing. The transmission probability might depend on several factors, e. g., the distance between the sensor and the measured event (shadow fading) [102].

Within our context, vehicles are the considered sensing device. Vehicles are able to percept knowledge about their surroundings, by the use of a variety of sensors. Gathered information is potentially of interest for other vehicles in the direct surroundings, as well as in larger distances, approaching the sensing position. Sharing this information enables neighboring vehicles to extend the size of their local perception [103]. To realize this information exchange, V2V communication has been introduced, which we have explained in Section 2.4.2.

Use cases, in focus since the introduction of V2V communication till today, are mostly safety systems like, e. g., collision warning or even collision prevention systems. Such systems are commonly based on direct single hop ad-hoc communication. However, several use cases, as introduced in Chapter 4, need an extended view. Several approaches exist to extend the transmission range of the sensing model [97]. For example, Lee et al. developed different storage architectures to store the event information for a longer time [104]. The Mobility-Assist Storage approach works without remote hardware to store event information. Collected information is transmitted when two vehicles are in transmission range. Using the mobility of the vehicles, the information is distributed within the network. In contrast to that, the Content-Addressed Storage approach keeps event information stored on remote servers and only propagates hash values, to identify the specific event via the vehicle network. However, V2V communication is in general limited in transmission range and poorly applicable, in case of sparse traffic.

An important mechanism in information collection is clustering of sensing sources, i. e., grouping the sources in geographical vicinity according to predefined rules [105]. The objective is to reduce the overall amount of data to be transmitted and thus, transmission costs. Data is aggregated at a so-called cluster head and collectively transmitted to the sink, e. g., a cloud service. According algorithms focus on cluster head selection, which might also perform data compression before transmission. The most basic cluster algorithm is named *lowest ID (LID)*. All nodes have assigned a unique *ID*, that is broadcasted to the neighboring nodes. Each node then allocates itself to the node with the lowest *ID*, that is implicitly selected as cluster head [106]. A variation is *highest degree clustering (HD)*. To minimize the amount of clusters, the head is selected based on the nodes with the highest number (degree) of nodes in direct communication range [107]. Another variation is the *weighted clustering algorithm (WCA)*, which is based on a performance indicator of several properties, like node degree, transmission power, mobility, and battery power [108]. An alteration of *LID* is *Lowest Relative Mobility Clustering Algorithm (MOBIC)*, designed for the use in vehicular networks [109]. Here, the *ID* is replaced by a performance indicator of node mobility (relative speed to neighbors). The *MOBIC* approach has been further devised within several developments. The *Distributed Group Mobility Adaptive Clustering Algorithm (DGMA)* uses a group mobility metric, named *linear distance based spatial dependency (LDSD)*, to prolong cluster lifetime [110]. This value describes the relative speed of two nodes and the movement direction. The *Adaptable Mobility-Aware Clustering Algorithm based on Destination positions (AMACAD)* extends the used metric by the destination, the speed and the current location [111].

The german automotive research project *CONVERGE* has introduced a different approach, that uses beacon-based clustering for performance improvements [112]. Network traffic is balanced by a probability based approach, to select the cluster head. Vehicles start with a probability of $p = 1$ and in a following adaption phase, neighbor vehicles communicate to decide to become the cluster head. Each time a vehicle receives such a message, the own probability to become the cluster head is lowered. The send-on-delta strategy is another approach to reduce data traffic [113]. Unnecessary data is discarded until a certain threshold, where performance depends on the size of the delta threshold and the change rate of measurements. An enhancement of this approach, that is based on the prediction of sensor values, is presented by Suh [114]. Linear prediction is used to estimate sensor values. Information is only

transmitted if an error threshold between the predicted and the measured sensor value is exceeded.

A transmission strategy based on a central sink node is *Ken*, developed by Chu et al. [115]. External queries are answered by the sink, that uses a predicted value of a replicated dynamic probabilistic model. The same prediction model is used at all sensor nodes. These send updates to the sink, if values differ from a certain threshold. This approach guarantees an accuracy within a certain range and the prediction outperforms linear progression of the send-on-delta mechanism. A system that collects correlating values, using a probabilistic model, is developed by Deshpande et al. [116]. A central server answers external queries and collects required information from the sensor nodes, similar to *Ken*, but false sensor value transmission is filtered out. In contrast to *Ken*, the server requests data if the uncertainty for a specific value is high. Thus, sensor nodes do not transmit data independently. This reduces the computational complexity on the sensor nodes, since these do not need a predicting function.

Another system, where also the server side requests for information, here named portal, and nodes are not allowed to transmit data independently, is named *CarTel* and developed by Hull et al. [117]. Here, vehicles are considered as sensor nodes. The system is based on opportunistic wireless connectivity, e. g., available Wi-Fi on driving by. Vehicles transmit data either directly to the portal or to other available devices, later delivering the data to the portal. Data is only transmitted once it was externally requested. Mobile applications directly query data from the portal. The portal requests the mobile nodes, i. e., vehicles, to send the required data to the portal server.

Local pre-processing is a general approach to reduce the amount of transmitted data. Typically data is compressed, to reduce the data traffic. An example framework, that utilizes data sparsity of the collected information is introduced by Li et al. [118]. They use a nonlinear algorithm to reconstruct the compressed data and an algorithm to perform random sampling on a sparse basis. Only a subset of samples is required to reconstruct all information without loss. This information collection theory is named Compressed Sensing (CS) and the used property of the sampled information is named sparseness in the transformation process. The proposed framework is designed to compress the information before sending in the IoT context.

3.4 THE ELECTRONIC HORIZON

The concept of a map based electronic Horizon (eHorizon) has been introduced in 2006, as a result of the *PREVENT* research project [119, 120]. The idea was to provide ADASs with a virtual map-based sensor, as information source beyond the local sensor range. The basic component is the local *horizon provider*, that serves as single source for all components and ADAS systems within the ego vehicle. The *horizon provider* knows the current position, velocity, driving direction and has also access to a digital road map. Based on this information, an eHorizon is constructed and provided as incremental stream via the ADASIS protocol on the vehicle data bus, typically the *CAN* bus [71]. The eHorizon consists of information ahead of the ego vehicle. Components and ADASs that require respective information, listen to the data bus [121]. Each of these components owns a *horizon reconstructor*, that incre-

mentally creates a 1D preview of the current vehicle track. The ADASIS protocol is developed by the *ADASIS Forum*[1], that is coordinated by *ERTICO*[2] and consists of most automakers and major suppliers [120]. In principle, any location based information can be encoded via this protocol. The latest version 2.0.3 of the ADASIS protocol has been released in December 2013. Map information is described as *paths*, connected with *stubs* (i.e. crossings) [24]. A *path* can consist of several *segments*. The basis eHorizon consists of the Most Probable Path (MPP), optionally with attached side paths. The prediction of the MPP is an essential component with a significant influence to the eHorizon quality [6]. The MPP can be derived from navigation system data, from historic mobility data or by a generic algorithm, based on attributes like road geometry [122]. In case of historic data, driven routes can be stored in a *past experience processing* and respective information can be reused, combined with turn signal and vehicle status information, to estimate the MPP [6]. A similar approach is described by Engel et al., that uses historical driving information to correct a generated MPP [122].

Thomas et al. have presented the *ActMAP* framework as concept for 'the wireless distribution of incremental map updates' [123]. Moreover, the system constantly compares road segments from a map or eHorizon provider with sensed information. The authors explicitly name the necessity to update in-vehicle map data, to provide a correct eHorizon, which is essential for the quality of ADAS. Möbus et al. highlight the increasing importance of an up-to-date eHorizon, that becomes increasingly important for ADASs [5].

Horita and Schwartz consider the eHorizon and ADASIS protocol, with respect to autonomous driving [124]. The authors investigate future needs in the ADASIS protocol and ETSI *LDM* [38] specification, to reduce the amount of required data traffic. With respect to autonomous driving, an extension to a lane level connection representation and higher precision lane data would be required. The position information of dynamic objects, i. e., the ego vehicle and neighbor vehicles, should be extended by *lane IDs*.

3.5 DISCUSSION AND PROBLEM STATEMENT

Although in literature mostly considered applications and use cases are related to the field of safety improvement, the overall range is very diverse. Typically, many descriptions overlap and classifications do not perfectly match. Consequently there is a need to bring all use case descriptions together to get a complete overview and derive a respective harmonized categorization.

With regard to vehicular ad-hoc networks, channel congestion, caused by position beacons, is a well known problem. Although many research work addresses the congestion control problem, it remains an issue. Existing approaches typically adapt the beacon rate and transmit power, mostly based on network parameters, like CBR. As a result, advanced approaches like *EMBARC* are able to prevent channel congestion and minimize the PER, but still fully utilize the wireless channel. This is due to two conflicting aspects: First, information accuracy should be as high as possible, i. e., the position beacon rate is kept as high as possible. Second, the channel load should

1 http://adasis.org
2 http://ertico.com

be as minimal as possible, to ensure reliability, i. e., a low beacon rate is desired. But to enable our aim of an extended perception, we require information forwarding of dynamic vehicle position data, which requires sharing of available channel capacity. Promising approaches incorporate fairness, with respect to actual communication needs, but typically argue with different application needs [85, 95]. From our perspective of the extended perception, the individual needs are comparatively the same for all participants. After all, several ADAS applications will use the respective perception information. Zhang and Valaee argue with different requirements for the individual vehicles [125]. We follow this argumentation, but see these requirements in the necessity of position broadcasts. The most effective control parameter is the message rate [126]. Thus, we see the major potential in the improvement of tracking prediction, to reduce the overall need to broadcast position beacons. Moreover, if the overall broadcast rate can be significantly reduced, an additional congestion control might be no longer required. According to Shladover and Tan, a tracking accuracy of 0.3m lateral and 0.5m longitudinal would be required in case of a collision warning application [127]. However, this is above a typical GPS accuracy and moreover, such critical systems must be based on local sensors. Unreliable wireless communication should not replace local sensors. Moreover, it can never be assumed to have a 100% penetration rate. Since we are focusing on a long-range perception for less safety relevant applications, a much larger tracking error might be tolerable.

With regard to sensor data collection approaches, the range of available approaches is very wide. However, mechanisms related to the domain of sensor networks are mostly not applicable, because of the high mobility in the vehicular domain. Vehicular data collection approaches are mostly based on clustering, to realize an optimized data transmission, towards a backend or the cloud. Due to the monetary costs for cellular network based data traffic, optimization is related to the amount of data transmitted. However, such a clustering approach requires participating vehicles to be equipped with ad-hoc communication capabilities. Moreover, a clustering approach assumes a required data transmission from several participants within a cluster, that can be aggregated, else it would just result in forwarding. In case of continuous sensing, e. g., temperature collection, approaches based on the predictability of prospective sensed data are suitable. Other strategies of incomplete data transmission focus on static network topologies. The highly dynamic nature of the vehicular domain requires an approach, that adapts to traffic density and ensures a maximum detection latency. In addition, it should be an incomplete transmission model to reduce the total amount of data traffic to a possible minimum.

The virtual sensor *eHorizon* has shown to be able to improve or even enable completely new ADAS applications. Moreover, the eHorizon becomes increasingly important for future ADAS, in particular with respect to autonomous driving [5, 128]. Besides some optimization needs, with regard to in-vehicular data dissemination, a major issue is potentially outdated map material.

As a result, we identify four major required steps towards our overall aim of an extended vehicular perception: First of all, relevant use cases have to be structured into a single consistent classification scheme. Second, an efficient mechanism is needed to gather relatively static vehicular sensed event data, independently of traffic density. Third, a mechanism is required to provide a long-range view in driving direction, based on previously gathered information. Fourth and finally, a mechanism with a

strongly reduced position beacon rate has to be developed, to include highly dynamic vehicle data into the long-range vehicular perception.

4 CATEGORIZATION OF USE CASE DEPENDENT INFORMATION DEMANDS

Interconnecting vehicles with each other or with infrastructure components causes different technical demands, depending on the respective use case. In state of the art literature many use cases and information demands are described. However, one can find many different categorization schemes, most according to the level of dynamic of the respective objects or attributes. To overcome this, we worked on a homogenized scheme, because each category brings its own technical requirements.

We have derived three zones of information demands from the use case descriptions of the ETSI specifications, standards of ITS vehicular communication and V2X applications, as well as relevant research publications [5, 8, 15, 38, 65, 68, 69, 70, 61, 66, 129]. Out of these use cases, we categorize three zones of information demands, namely *safety zone*, *awareness zone* and *information zone*. We present our categorization scheme in [130] and present a revised and extended version in the following. The distinction of the zones belongs to the tolerated latency between the occurrence of an event and the point in time, the information has to be processed at the receiver side. Categorizations regarding property or object dynamics are also mapped to this latency related schema. However, there is no sharp boundary between these zones and many events affect more than one single zone [131]. For example, if a vehicle breaks down, a safety application can carry out an evasive brake, a lane change, or evasion maneuver and this can be assigned to the safety zone. At the same time, a route guidance application can suggest an alternative route and this can be assigned to the information zone. In this chapter, we first discuss aspects regarding the communication range in urban and highway scenarios. Afterwards, we describe these three zones of information demands in more detail.

4.1 VEHICULAR COMMUNICATION RANGE

To discuss the technical demands of the three different zones of information demands, it is a prerequisite to discuss the potential communication range of vehicular communication. Fundamentals about vehicular communication have been introduced in Section 2.4. In the following, we will look into details about the communication range of vehicular ad-hoc communication.

The most influencing factors with regard to the communication range are the used frequency and the maximum transmit power. As described in Section 2.4.2, vehicular ad-hoc communication is allocated in the 5.9 GHz. The control channel is dedicated to IEEE channel number 180, with a center frequency of 5.9 GHz, in the EU and in the US to IEEE channel number 178 with a center frequency of 5.89 GHz [27, 29]. The Free Space Path Loss (FSPL) in decibel is defined as $\mathrm{FSPL(dB)} = 20\log_{10}\frac{4\pi df}{c}$. To be able to assess the influence of the distance, with regard to the received signal strength, the free space loss for frequency $f = 5.9\mathrm{GHz}$ is given in Table 2. The transmit power for the control channel is limited to 33 dBm EIRP (Equivalent Isotropically Radiated

Table 2: Free space path loss for an isotropic radiator at $f = 5.9\text{GHz}$.

Distance [m]	1	10	100	200	300	400	500	700	1000
dB	47.86	67.86	87.86	93.88	97.40	99.90	101.84	104.76	107.86

Power) [27]. This results in a theoretic maximum signal strength at the receiver side of -74.86 dBm in a distance of 1000 m, only considering free space path loss without any other disturbances. The required Signal to Noise Ratio (SNR) at the receiver side depends on the used data rate. According to the European ETSI ITS G5 specifications, using a simple path loss model and a transmit data rate of 3 Mbit/s, a maximum transmission range of 1000 m is possible [40]. But the default data rate for the control channel is 6 Mbit/s, that requires a 3 dB higher SNR. This data rate has also shown to be the best for the considered communication ranges and message sizes [132]. Moreover, the transmit power is down-regulated till 10 dBm by the Transmit Power Control (TPC) of the used DCC mechanism, to reduce collisions on the wireless channel, which reduces the transmission range [133].

However, wireless transmissions are influenced and attenuated not only by FSPL but also by shadowing (i.e. obstructions in the line-of-sight), fading (i.e. multi-path propagation causes interference), interference and diffraction (i.e. deflection of waves at the edges of objects), reflections (i.e. at the surface of obstacles) and Doppler shift (i.e. a frequency shift of the signal) [134]. Communication range can be drastically reduced by obstacles in the environment, especially in urban environments as shown by Sommer et al. [135]. They have modeled buildings and respective properties of signal propagation, based on an empirical model. The model handles shadowing effects in VANETs and abstracts from diffraction and reflection effects. They have shown that the number of times an obstacle border is intersected and the length of the obstacle intercession has a huge attenuation impact that can be modeled by a certain factor. For the majority of observations, they parametrized the attenuation to 9 dB per wall and 0.4 dB per intersected meter. Real world measurements during the evaluation have shown the high accuracy of the model. In general, outside of city environments the major source of signal variation and attenuation are vehicles and in urban scenarios the major source are buildings and vegetation, but vehicles are here also an important source of obstruction [136].

A similar but more detailed model is used by Boban et al., that uses outlines of vehicles, buildings and foliage [136]. For signal propagation basically three models are conducted: line of sight, non line of sight, caused by vehicles and non line of sight, caused by static obstacles. They have gathered a large set of real world measurements in Porto and Pittsburgh for model evaluation. Their measurements show in an open space scenario, with line of sight communication, an average receive power of -90 dBm in a distance of about 420 m. On highways, they have measured an average receive power of about -85 dBm in a distance of 300 m. In urban scenarios this decreased to about -80 dBm in a distance of 100 m. This is mainly because transmission through buildings has a very high attenuation and is much less important than diffraction and reflections around buildings [136, 137].

All this infers that a simple path loss model, like used in ETSI technical specifications, is not sufficient [40]. Jiang et al. have investigated the packet reception rate for different message sizes and different vehicle densities per km [132]. They have

Table 3: Overview of velocity dependent communication range in an urban scenario with an assumed V2V communication range of 100 m. To provide information 5s or respectively 30s in advance, the transmission needs several hops between sender and receiver, depending on the velocity.

Velocity	$[\frac{km}{h}]$	$[\frac{m}{s}]$	$\delta_s(5s)$ [m]	hops	$\delta_s(30s)$ [m]	hops
	30	8.33	42	1	250	3
	50	13.89	70	1	417	5
	70	19.44	98	1	584	6

shown that the packet reception probability decreases to about 70% in half distance of the theoretic maximum communication distance, in a scenario with a vehicle density of 400 vehicles per km and a message size of 200 Bytes. An increase of vehicle density or message size further decreases package reception probability.

Mahler et al. have simulated the packet reception rate based on channel sounder measurements for two highway scenarios with different traffic density [138]. They have investigated the packet reception rate for oncoming vehicles and also evaluated different message sizes. The results show a clear drop in the packet reception rate in a distance of about 400 m.

Meireles et al. conducted measurements about the impact of vehicular obstructions in three different road scenarios: urban, suburban and highway [139]. Measurement results show a high packet reception rate in line of sight scenarios, even for long distances, but a clear drop at about 400 m distance for some scenarios. In case of non line of sight, caused by other vehicles, in a distance of about 100 m, the packet reception rate is decreasing and for some cases down to zero in a distance of about 500 m. In a suburban environment the Received Signal Strength Indication (RSSI) decreases to 10 dB in about 100 m distance for non line of sight, caused by other vehicles. In an urban canyon the RSSI decreases to 10 dB in about 200 m distance, due to the tunneling effect with reflections of relatively low phase difference. In the highway scenario a decrease of the RSSI to 10 dB is in about 300 m.

An extensive field test has been conducted within the german research project initiative Ko-FAS[1]. Paschalidis et al. have investigated urban street intersections by real world measurements [140]. They have shown a high attenuation for crossing traffic scenarios at intersections for non line of sight communication. Other measurement results within KO-FAS show a reliable line of sight communication at roundabouts of about 400 m. Mahler et al. show that at intersections a two dimensional distance measure is necessary, i. e., setting the packet delivery rate in relation to each vehicle's distance to the crossing center of the intersection [141]. They show that non line of sight communication strongly dependents on obstructing buildings and possible reflection surfaces. If reflection surfaces are available, non line of sight packet reception rate is reliable in a distance for each vehicle of about 50 m, or a relative distance of 100 m.

Based on the above-mentioned findings of inter vehicular communication in the 5.9 GHz band, we assume in the following a reliable V2V communication range of 300 m for highway scenarios and 100 m for urban scenarios. Based on this we investigate

1 http://ko-fas.de

Table 4: Overview of velocity dependent communication range in a highway scenario with an assumed V2V communication range of 300 m. To provide information 5s or respectively 30s in advance, the transmission needs several hops between sender and receiver, depending on the velocity.

Velocity	$[\frac{km}{h}]$	$[\frac{m}{s}]$	$\delta_s(5s)$ [m]	hops	$\delta_s(30s)$ [m]	hops
	80	22.22	111	1	667	3
	100	27.78	139	1	833	3
	120	33.33	167	1	1000	4
	130	36.11	181	1	1083	4
	200	55.56	278	1	1667	6

use case dependent communication needs in the following sections. From the use case perspective, we consider the distance from the ego vehicle to the point of interest. Table 3 gives an overview of typical vehicle velocities (30, 50 and 70 km/h) in an urban environment in relation with the respective driven distance within 5 s and 30 s. These two periods of time are related to the three zones of information demands that are discussed in detail in the following sections. Moreover, the minimum necessary communication hops, that are necessary to bridge the respective distance, are given. Table 4 gives an overview of typical vehicle velocities (80, 100, 120, 130 and 200 km/h) in a highway scenario in relation with the respective driven distance within 5 s and 30 s. Also, the minimum necessary communication hops, that are necessary to bridge the respective distance, are given. As mentioned before, the following sections discuss the three different zones of information demand.

4.2 SAFETY ZONE

The most popular use cases regarding interconnected vehicles are related to collision prevention or collision risk warning. Here, we are dealing with highly dynamic information and high latency demands. These use cases are related to our first use case zone, namely the *safety zone*. In the ETSI technical report of application collec-

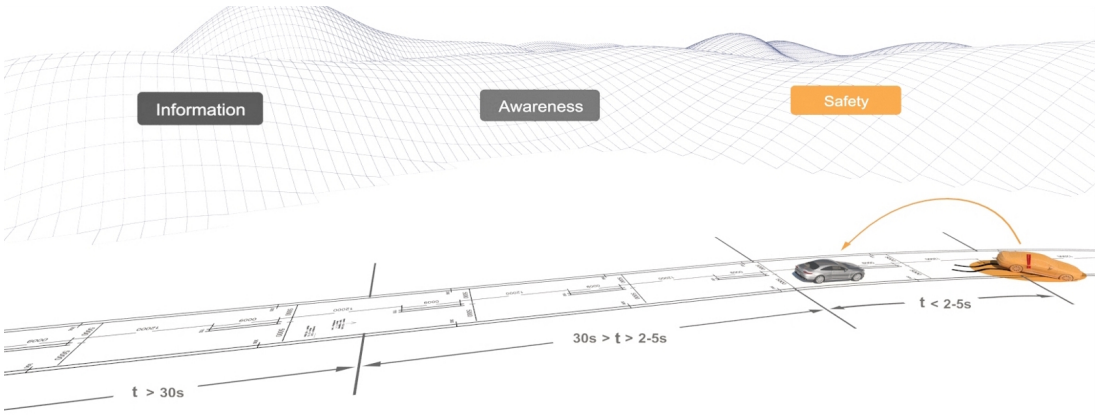

Figure 4: Illustration of the *safety zone*.

tion, these use cases are categorized as 'active road safety' [68]. The document also incorporates contributions from the EU research project PRE-DRIVE C2X and the car-2-car communication consortium working group[2]. Basically we aggregate all described safety relevant use cases into this one *safety zone*. This includes also use cases that are defined to be executed automatically because of the low response time and in some cases only to mitigate the damage caused by collision. At the end we stick to the ETSI time definition of $2-5$ s as boundary of this zone [69]. This defined latency of $2-5$ s is the complete time from the occurrence or sensing of the event, processing on the sender side, communication, processing on the receiver side till the start of the respective action. Basically, this zone has two main subcategories, namely collision prevention and collision risk warning. An example for collision prevention is a merging traffic turn warning, where the presence, the exact position and movement of incoming vehicles is propagated via the communication channel. The receiving vehicle can react immediately to prevent a lateral collision. An example for collision risk warning is the emergency electronic break light, where the emergency break event is propagated via the communication channel to enable an automated braking of succeeding vehicles with low response time. A detailed description of all use cases is given in Section A.1.

From the communication perspective, direct ad-hoc communication with broadcast of messages and no connection establishment, as common in VANETs, is here preferable. This is due to the high latency demands and the fact that the availability of cellular networks can not be guaranteed within the short communication window. It can be seen from Tables 3 and 4, that using IEEE 802.11p based vehicular ad-hoc communication, within this *safety zone* the ego vehicle should be always within direct one hop communication distance. It should be mentioned that in general higher speeds are driven on more straight roads that will allow rather higher communication ranges than assumed in Tables 3 and 4.

Message content is mainly related to object position, velocity, acceleration, heading, steering angel, yaw rate or even about an emergency event like an occurred collision or emergency break. An illustration of *safety zone* related communication is depicted in Figure 4. The vehicle is close to the point of a possible collision or emergency situation. The remaining driving time to this point is less than $2-5$ s. Due to the low latency demands of these use cases it is not enough time to give an alert signal to the driver, because the driver's reaction time will use up most of the remaining time. Thus ADAS systems addressing these use cases have to react autonomously, e.g., execute an emergency break or perform an evasive trajectory.

Basically, these use cases have been focus in research for several years. The respective communication solutions are content of the European ETSI C-ITS G5 and the US WAVE standards. Thus these are not part of the work at hand, and we focus on the following zones described in Sections 4.3 and 4.4.

4.3 AWARENESS ZONE

The awareness zone describes use cases that require immediate attention or response of the driver. Prominent use cases are cooperative driving maneuvers and use cases that make use of information about moving objects in further proximity. Here we are

2 https://www.car-2-car.org

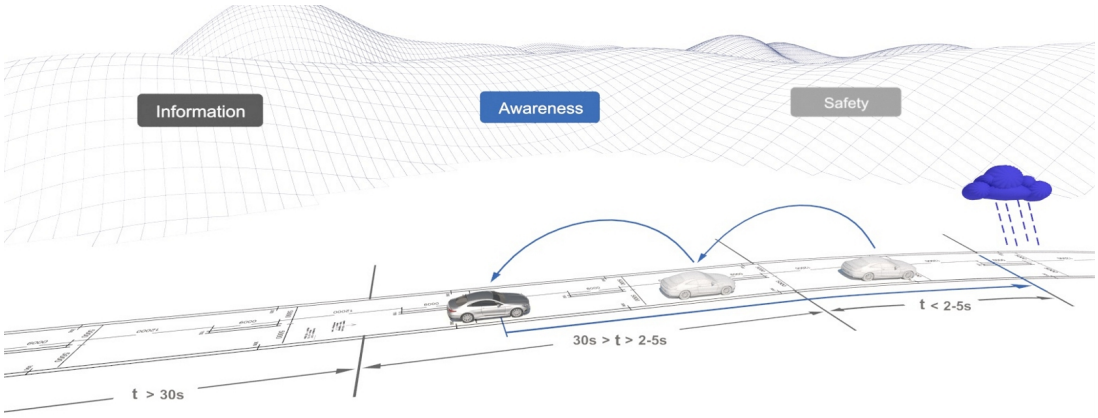

Figure 5: Illustration of the *awareness zone*.

dealing with dynamic information in non safety critical situations. For definition of the time boundaries we stick to the ETSI time definition according to the *safety zone* [69]. This results in a lower boundary of $2-5$ s and an upper boundary of 30 s, each with respect to the remaining driving time till the point of interest.

In particular, many of the use cases need dynamic information about moving objects, i.e. other road users, in a relatively long distance range. This is especially for cooperative driving maneuvers. Here, potential cooperation partners have to be identified and a slack time is needed for the acknowledgement of the human driver. The human driver can always oversteer any maneuver and take over control, which implicitly refuses the started maneuver. Another example is cooperative adaptive cruise control to reduce traffic jams and fuel consumption. Long-range cooperation is necessary for creating a stable traffic. This can only be realized by cooperation based on communication. The single use of local sensors is not sufficient to prevent concertina effect caused traffic jams.

Unfortunately cooperative maneuver use cases did not get much attention in research so far. But this has changed, since a lot of research and field tests in the area of vehicular communication, as well as partly or highly automated driving, have shown future needs of driving automation towards autonomous driving. This issue is also a basic component of the current research and innovation framework *New Vehicle and System Technologies* of the german Federal Ministry for Economic Affairs and Energy[3]. The program explicitly highlights the need of communication technologies to enable fast, save and reliable cooperation between vehicles. It requests technologies to extend the vehicular perception range. Within the mentioned program, in September 2016 the project *IMAGinE* (Intelligente Manöver Automatisierung - kooperative Gefahrenvermeidung in Echtzeit) has started with a consortium of all major national automakers and first tier suppliers[4]. This shows the importance of technologies, enabling applications according to the *awareness zone*, for the realization of highly auto-

3 Research and innovation framework *New Vehicle and System Technologies*, Federal Ministry for Economic Affairs and Energy (BMWi), `http://www.tuvpt.de/fileadmin/downloads/bmwi_Neue_Fahrzeug-_und_Systemtechnologien_2015_s06.pdf`, last accessed and validated in February 2017

4 Press release, 'BMWi startet Förderung des Großforschungsprojekts IMAGinE für kooperatives Fahren in der Zukunft', `http://www.bmwi.de/DE/Themen/technologie,did=779040.html`, last accessed and validated in February 2017

mated driving and future autonomous driving. A definition about different vehicle automation grades is given in Section 2.2.

From a communication perspective, multi-hop ad-hoc communication or cellular communication are feasible technologies. Latency demands are not as high as for safety applications. From the perspective of the remaining distance to drive, there is a lot of time for information propagation. But information latency demands are still relatively high since information content in these use cases is also dealing with dynamic object information of moving objects. The exact time tolerance is use case dependent and not generally settleable. But in general, position information of dynamic objects should not be older than two or three seconds, to ensure a certain accuracy in position estimation. Moreover, the availability of cellular communication can not always be guaranteed, which makes ad-hoc communication more preferable. It can be seen from Tables 3 and 4, that using IEEE 802.11p based vehicular ad-hoc communication within the *awareness zone*, vehicles will need multi-hop communication of up to six hops to propagate dynamic object information.

Message content is mainly related to dynamic object information, i. e.position, velocity, acceleration, heading, etc., as a list of all objects within the perception range. An illustration of *awareness zone* related communication is depicted in Figure 5. A vehicle is interested in dynamic object information and moreover in sensed environmental information, which is the rain drop rate in this example. One vehicle is driving in the distance of interest. The remaining driving time to the area of current interest is beyond $2-5$ s, but less than 30 s. The ego vehicle is not in direct ad-hoc communication range, but in two hop communication range with another vehicle in between. Information is broadcasted by the foremost vehicle. The intermediate vehicle aggregates the received information with own sensor data and then also broadcasts the information. The ego vehicle, i. e., the rearmost vehicle, is now able to receive this information and can aggregate the received information into the own perception model. Thus, a communication mechanism, enabling long-range perception, has to make use of multi-hop communication. Here, channel load has to be minimized since congestion becomes rapidly a problem for higher traffic densities [142, 143, 93]. However, if the communication mechanism is very efficient and network coverage is good, then also hybrid communication could be possible.

As mentioned before, these use cases have not been much in focus within research and standardization. Message types and information propagation mechanisms have to be further developed. These use cases are focus within this thesis and appropriate technologies will be developed within Chapter 7.

4.4 INFORMATION ZONE

The *information zone* describes use cases that require no direct attention or response of the driver. Information updates typically affect the route selection or route changes. Prominent use cases are traffic condition information or regulatory information, e. g., traffic sign information. Here, we are dealing with static or semi static information in further driving distance. These use cases are typically not safety relevant and focus on driving comfort and economy. For definition of the time boundaries, we adopt the ETSI time definition according to the *information zone* [69]. This results in a lower boundary of 30 s, with respect to the remaining driving time till the point of interest.

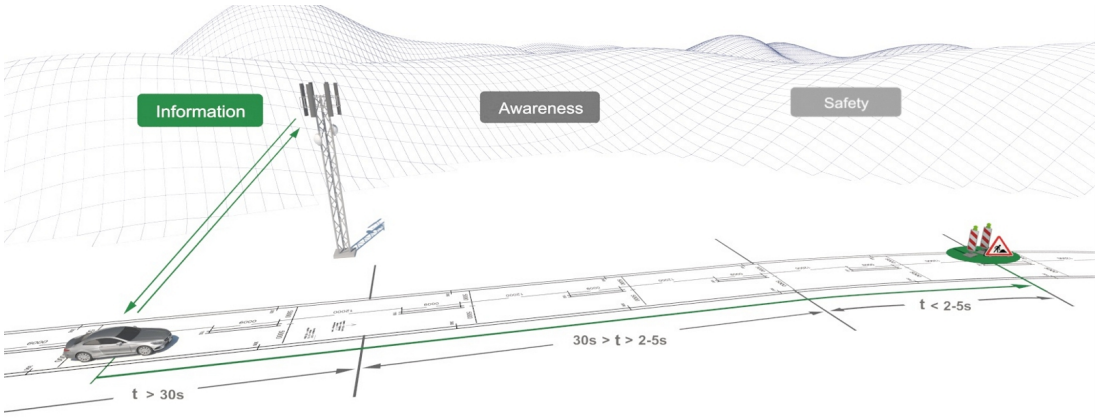

Figure 6: Illustration of the *information zone*.

In particular, the respective use cases deal with static or semi static information that is flow influencing, about road segments or basis of Location Based Services (LBS). Driving goal within these use cases is typically to reduce fuel or energy consumption and to increase driving comfort. Information exchange has to fulfill only very low latency demands, but information has to be at the respective vehicle a long distance in advance. This is necessary to enable very predictive driving, or even rerouting, within these use cases.

From communication perspective, multi-hop ad-hoc communication or cellular communication are feasible technologies. There is a lot of time for information propagation, in perspective of the remaining distance to drive. It can be seen from Tables 3 and 4, that using IEEE 802.11p based vehicular ad-hoc communication within the *information zone*, vehicles will need multi-hop communication of many hops to propagate information within such a long distance. Due to the relative long distance between the area of interest and the ego vehicle position, the probability is high that not enough other vehicles are in between for message forwarding. This applies especially outside of bigger cities or at night. On the other hand the availability of cellular communication can not always be guaranteed. But due to the relatively long driving distance, it is very unlikely that cellular connection will not be available for a longer duration. Moreover, data regarding the information zone is more static, rather than dynamic. This means that information can in general be transmitted to the vehicle a lot more in advance than 30 s prior approaching the point of interest. Due to this, cellular communication is a lot more preferable for use cases according to the *information zone*.

Message content is mainly related to static object information, e. g.about traffic signs including the respective position and range of validity. An illustration of *information zone* related communication is depicted in Figure 6. A vehicle is interested in information along the planned driving route and requests respective information from a backend via cellular communication. The remaining driving time to the area of current interest is beyond 30 s. The requested information is along the route in a longer distance and other vehicles are not always in between. In such a situation, mobile ad-hoc networks will not work, except a dense availability of infrastructure side communication units, named RSU. But by the use of cellular communication, a backend can keep the information available, transmitted by a vehicle that previously

passed the area of interest. However, since cellular communication is costly, strategies for data collection as well as information dissemination have to be chosen wisely. The total amount of transmitted data has to be minimized.

As mentioned before, these use cases have not been focus in research of connected vehicles. Message types, data collection strategies and information propagation mechanisms have to be further developed. These use cases will be focused within this thesis. Appropriate technologies for information collection will be developed in Chapter 5 and for information dissemination in Chapter 6.

4.5 CONCLUSION AND SUMMARY

Within this fourth Chapter, we have introduced our scheme of three zones of use case groups, depending on the respective information demand, namely *safety zone*, *awareness zone* and *information zone*. We have analyzed different use case descriptions and categorization schemes from the use case descriptions of the ETSI specifications, standards of ITS vehicular communication and V2X applications, as well as relevant research publications. We have harmonized and restructured these use cases into the previously named three zones of information demands. Moreover, we have analyzed vehicular ad-hoc communication range in literature. Based on the gained knowledge we have derived the feasible communication technology. A more detailed overview and description of the single use cases belonging to each zone can be found in Section A.1.

5 GATHERING VEHICULAR SENSED DATA

Direct ad-hoc V2V communication is designed to share information between vehicles in direct proximity. However, several ADAS use cases, especially in the domain of predictive and economic driving, need information along the driving path, also in long-range. Due to the nature of long-range communication, ad-hoc communication would require forwarding of information over many hops. Especially in situations of sparse traffic, this approach would fail. Thus, an infrastructure support would be necessary, which could be either RSUs or cellular networks. Since an extensive expansion with V2X infrastructure units, i. e., RSUs, is very unlikely in the near future, a cellular based approach will be considered in the following. A respective approach can benefit from an already existing high cellular network coverage. Moreover, a single central, at least logically central, up-to-date information base could provide consistent information to all participants. But the number of potentially connected vehicles to such a system is very large. This causes the need of an intelligent data collection management, that is able to minimize data traffic and thus, minimize transmission costs. In this chapter, we present a system to gather vehicular sensed data with a predefined detection latency and reduced wireless communication costs. The according use cases are related to the *information zone* as described in Chapter 4.4. A possibility to provide such long-range information along the driving path is the cloud based *eHorizon*, as presented in the next Chapter 6.

The remainder of this chapter is structured as follows: In the next Section 5.1, we describe the need of an according system and our objectives, followed by the description and overview of our system for probabilistic sensing in Section 5.2. In the subsequent Section 5.3, we provide our evaluation. We begin with a description of our evaluation setup, followed by the evaluation metrics. Finally, we discuss our evaluation results and findings.

5.1 PROBLEM STATEMENT AND OBJECTIVE

Collecting vehicular sensed information, with the objective to provide a long-range information view to other vehicles, is the basis to enable ADASs with regard to the *information zone*, as introduced in Chapter 4.4. Today, introduced systems focus on single use cases to cope with scaleability. A typical application is traffic flow measurement, where connected vehicles and devices send a movement status to a backend. In general, cellular communication is costly, since the used frequencies are commercial and usage has to be paid, as mentioned in Section 2.4.1. Moreover, in the domain of vehicular sensing, a system has to deal with millions of clients. Thus, a more generic approach would require an intelligent management, to reduce communication costs and ensure scaleability.

Classic data collection approaches focus on optimizations in the field of opportunistic sensing, i. e., a complete data transmission. Used mechanisms are local preprocessing on the mobile client or clustering with aggregation, to reduce data traffic.

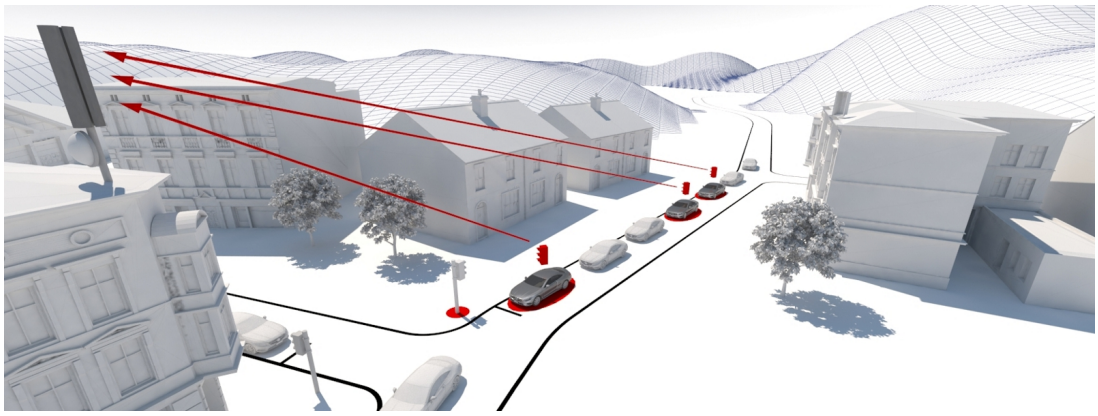

Figure 7: Use case example of traffic light status recognition with probabilistic transmission.

Another possibility is an incomplete transmission model. Due to the potentially high number of mobile sensors, in terms of connected vehicles, a complete transmission would cause a high redundancy. Typical use cases are the detection of changes within the road network, including a change of traffic signs. An example is given in Figure 7. Several vehicles are waiting and form a queue at a red traffic light. A simple map, with information about the position of traffic lights, would be sufficient to detect this status by the proximity to the traffic light and the lack of movement. Assuming all vehicles are equipped with a connectivity module, the receiver side would get the same information several times. A specific redundancy, defined by the respective use case, would be required to ensure reliability and trust. Every transmission above this required level would be unnecessary redundancy. Thus, our aim is to design a mechanism that is able to reduce the data traffic of this unnecessary redundancy. Moreover, the system should still fulfill certain QoS metrics, e. g., the maximum detection latency. This detection latency is the time since the event occurred, e. g., a new traffic sign, until the minimum required number of event detections is transmitted. In our case, we assume a transmission to a central backend, to serve as data source for a long-range information provision. The design of such a backend is not in focus of this work, but a logically central backend system does not necessarily need to be a physically centralized structure.

Our basic concept is a probabilistic data collection strategy. By introducing an event type-based transmission probability, the amount of transmissions can be reduced. Especially in dense traffic situations, an event might be sensed very often. The resulting redundancy can be reduced by introducing transmission probabilities. In our approach, we control the transmission probabilities separately for each event type. Since vehicular traffic is subject to spatio-temporal changes, transmission probabilities also have to be continuously adapted. We conduct the management, based on geographic cells, since the area of adaption should have preferably a homogeneously distributed traffic.

Another example to illustrate our approach is the detection of traffic signs, either new ones or changes, e. g., in variable traffic signs. An according illustration is given in Figure 8. If the penetration rate with communication and sensing capabilities is high enough, then the event, i. e., a specific traffic sign detection, would probably be sensed several times, within a relatively short period of time. Vehicles sense the traffic sign on passing by and then roll the dice if the sensed information should

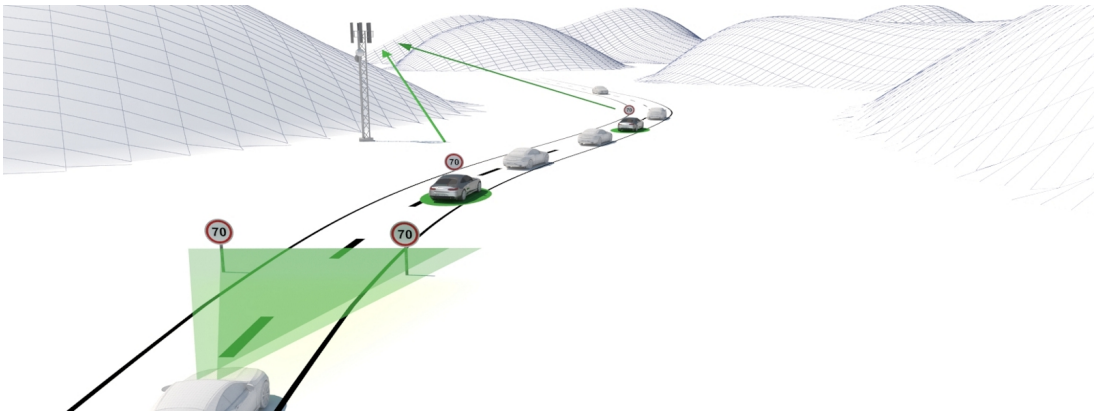

Figure 8: Use case example of traffic sign recognition with probabilistic transmission.

be transmitted, according to the given transmission probability. As a result, only a subset of all vehicles will transmit the according data.

Regarding the type of detected information, we distinguish between two general types, namely discrete und continuous events, whereas this naming is adapted from [144]. Discrete events are location based, but we do not know where they occur. A representative example is the detection of a traffic sign or route diversion. Moreover, we assume to have no knowledge about the routes of the vehicles. Thus, we have mobile sensors with unknown movement and no knowledge where the events to sense occur. Continuous events are not stationary location based, but continuously detectable. Representative examples are sensing the temperature or the rain drop rate. Here, a certain sensing density, i. e., a number of sensing gatherings per area and time, might be desired. Both types, discrete und continuous events, can vary over time.

Our developed model uses the detection latency, i. e., the time it takes to gather the required redundancy of a discrete event, and data density, i. e., the number of sensing gatherings per square kilometer and hour, as QoS boundaries. Based on this, the backend-based management component calculates the required transmission probabilities, to adapt to the respective setpoint. By down-regulating the transmission probabilities, we attempt to receive exactly the minimum required number of event detections. As a result, we are able to considerably reduce the received redundancy and accordingly the number of transmissions is reduced.

We have named our probabilistic data gathering approach *ProbSense.KOM*. We have presented this approach in our previous work [96] and present a revised and extended version in the following. The extension of *ProbSense.KOM* uses a hybrid communication approach of direct ad-hoc communication and cellular networks. We have named this extended approach *Hybrid-ProbSense.KOM*, that we introduced in [145] and present a revised and extended version in the following. In the next Section 5.2, we give a detailed description of our system of probabilistic gathering vehicular sensed data. We start with the probabilistic approach of *ProbSense.KOM*, followed by the extension of *Hybrid-ProbSense.KOM*.

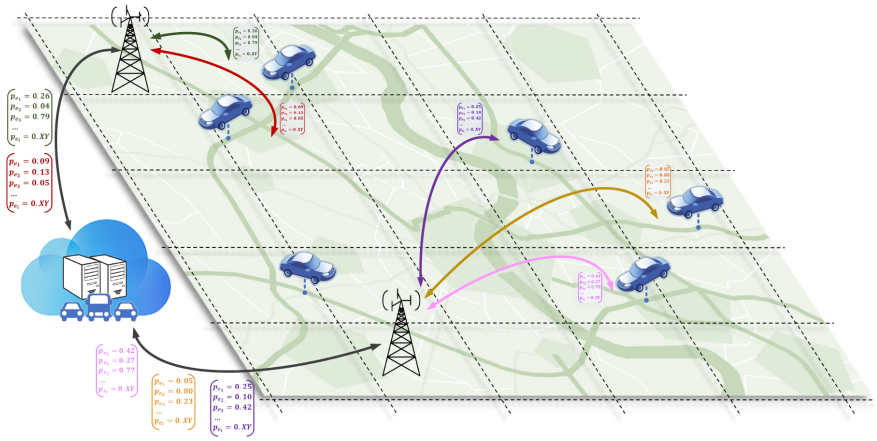

Figure 9: System overview of ProbSense.KOM [96].

5.2 SYSTEM CONCEPT AND SPECIFICATION OF DYNAMIC PROBABILISTIC SENSING

The objective of our system is to provide the means to gather vehicular sensed information via cellular communication technology. This information is the foundation for a central up-to-date map database. Since cellular communication is potentially costly, as mentioned in Section 2.4.1, communication should be minimized. In our model, we assume that a certain detection latency can be tolerated at the backend. We use an incomplete transmission model and gathered information is only transmitted with a certain probability.

Our generic approach basically consists of two levels: In the first level, i. e., *ProbSense.KOM*, we assume vehicles to serve as mobile sensors, with only cellular connectivity available. All vehicles maintain a TCP connection to the backend. The communication between the mobile clients and the backend is realized via a MQTT[1] based publish subscribe system. This guarantees a scalable connection management and brings an asynchronous communication, which decouples the bidirectional information flow between the mobile clients and the backend. This is especially beneficial in case of varying connectivity. Our system concept is visualized in Figure 9. The map is subdivided into geographic cells, or just geo cells, that will be discussed in more detail, later in this Section. At the backend, the number of vehicles currently in each geo cell, is known and incoming data is measured. The backend calculates the transmission probabilities individually for each geo cell and for each event type individually. The calculation is based on knowledge about the number of vehicles per geo cell and the amount of incoming messages. Transmission probabilities are propagated to all vehicles, individually per geo cell, as an according probability matrix. This controls if a detected event is transmitted to the backend, i. e., which sensor data to be considered, and allows to adapt to different traffic densities, by adaption of the corresponding transmission probability.

The system controls against a predefined setpoint, i. e., tries to achieve a predefined incoming data rate, respectively detection latency. In case of continuous events, the setpoint is a data rate per area and time. In case of discrete events, the setpoint describes a predefined number of messages about an event, within a specific time.

1 http://mqtt.org

The latter describes the detection latency, the system needs to identify the discrete event. Before propagating new transmission probabilities towards the vehicles, the system calculates possible transmission savings and the transmission costs to propagate an update. The update is done only if the estimated savings are greater than the accruing control costs. The complete calculation process is described in more detail later in this section. If a vehicle detects an event, i. e., by sensing a discrete event on driving by or a time trigger for continuous events, it is decided if the sensed information should be transmitted, based on the respective transmission probability.

In the second level, i. e., *Hybrid-ProbSense.KOM*, we assume direct ad-hoc communication to be available, in addition to cellular communication. Thus, the second level is a hybrid communication approach. This second level works basically the same way, as the previously introduced cellular based first level. But in addition, vehicles broadcast detected events via direct ad-hoc communication. This broadcast is done always when an event is detected. Receiving vehicles keep this knowledge in memory. The difference of the hybrid approach is in case of a vehicle senses an event. In this case, it is also decided, based on the respective transmission probability, if the sensed information should be transmitted. But in case of a positive decision, i. e., the event information should be transmitted, the message also includes information about neighboring detections. This means, that the sender ID and timestamp, received as ad-hoc broadcast message from neighbors, is added to the event detection message. So the transmission decision process on the vehicle side remains the same, as described for *ProbSense.KOM*.

Before we continue with the formal description of our approach, we give some details about the concept of geo cells and the underlying probability distribution.
Geographic Cells: The concept of geo cells is a virtual segmentation of the map into smaller parts. As the shape for the geo cells we have selected a rectangle, since this shape is easy to describe. Simple forms to describe a rectangular are just two geo coordinate points, one geo coordinate point and a value for length and width, or in case of quadratic cells, just one geo coordinate point and a value for the width. Moreover, rectangular cells can be directly derived from the geo coordinates. An example is to divide each geo coordinate, i. e., latitude and longitude, with a predefined value and discard the modulo. The results are two integer numbers, that can be concatenated to an unique ID. Using this approach, every mobile node can directly calculate from its own geo coordinates, the ID of the geo cell it is currently in. However, also every other shape might be feasible. The geo cell is just the unit of the area to control the transmission probabilities. In general, not the shape, but the size of the geo cells is important. As it will be shown later in Section 5.3, only vehicle homogeneity, within a geo cell, is desired. Thus, the size of the geo cells has to be small enough, to enable an optimized adaption of the transmission probabilities. But if the cell size is getting to small, the overall transmission costs are increasing. This is due to the control traffic for updating the transmission probabilities of the vehicles, as result of an increasing number of cell changes.
Probability Distribution: The binomial distribution describes the probability of the number of successes in a sequence of n (number of trials) independent true/false experiments, each with a probability of success p. Hence, it describes the probability of k successes, in case of exactly n trials. The true and false experiment in our case is to send and not to send. However, we want to have a fixed number of successes k, i. e., our required redundancy, but do not know the number of trials n. Every received

message at the backend contributes to the number of successes k. If the number of successes k would increase, it would result in unnecessary redundancy. The total number of trials n is not known at the backend, because only k successes will be transmitted. The not transmitted trials, i. e., the fails, will not be transmitted and are thus, not countable at the backend. Hence, in our case we search the probability, to achieve exactly k successes. The binomial distribution describes the probability to achieve k successes, within fixed n trials. The negative binomial distribution describes the probability to need n trials to achieve fixed k successes. By the use of the set probability p_t, the transmission probability in period t, we can estimate the number of trials $n_{est,t}$ in period t, that caused $k_{obs,t}$ observed successes in period t. Finally, $n_{est,t}$ and the desired k can be used to calculate the required p_{t+1}, with the negative binomial distribution.

5.2.1 *Formal Specification*

We divide the whole map equally into virtual geo cells. Each of these geo cells $m \in M$ can be considered independent from the other cells. Thus, we also process the optimization independently and all parameters in the following, belong always to exactly one geo cell m. Processing is done in discrete time steps t, named adjustment period. The sum of all t since system start, is denoted by T. The set of all event types e is E. We define va_t as the total amount of vehicles in a geo cell in period t. The required detection redundancy, for the respective event type e, is defined as τ_e. The size of a data packet, containing information about a detected event, is given with x_e. The size of a control message, regarding the respective event type, is given with b_e. In each geo cell, a probability $p_{e,t} \neq 0$ determines, if a detected event of type e is sent. The total number of by vehicles sensed events of type e, within period t, is defined as $n_{e,t}$ and is not influenced by the transmission probability. We define $F_{tot(e,t)}$ as the total amount of data traffic produced in one specific geo cell for event type e in period t.

Discrete events are a subset $E_D \subseteq E$. The maximum latency $l_{max,e}$ is set for each discrete event. It defines the threshold after which the event should be in $\rho\,\%$ of all cases stored in the database. The latency of all events measured in the adjustment period t is defined as L_t. The $\rho\,\%$ quantile of this set L_t is denoted by $l_{e,t,\rho}$. If $l_{e,t,\rho}$ or more events are received within a latency less or equal $l_{max,e}$, then a minimum of $\rho\,\%$ events are received within the threshold.

Continuous events are also a subset $E_C \subseteq E$. For continuous events, the desired detection density is given as δ_e. The measured detection density in period t for an event of type e is $\delta_{e,t}$, i. e., the received events of type e, for the according geo cell m.

The weighting function, that quantifies the quality goals of the event data collection for the respective events of type e in period t, is given in Equation 7. In every period t and for every event type e, individually for each geo cell m, we try to minimize the product of the data traffic $F_{tot(e,t)}$ and a calculated weighting ϵ_e. The weighting ϵ_e is calculated depending on, if e is element of E_D or E_C. The subtraction of the measured and the desired quality metric is normalized by division of the desired quality metric. Deviation of the desired quality threshold is penalized by a quadratic function. We have not considered an exponential function, in order to prevent divergence, but a quadratic penalizing, which converges definitely. In case

Table 5: Overview and description of the used parameters. Since every geo cell m is considered separately, the respective parameter always belongs to the one geo cell m.

Parameter	Description
m	A single geo cell. $m \in M$
M	The set of all geo cells.
t	Adjustment period, i. e., the period of the processing. trigger at the backend.
T	The sum of t, since system start.
e	Event type.
E	The set of all event types e.
E_D	Discrete event types. $E_D \subseteq E$.
E_C	Continuous event types. $E_C \subseteq E$.
va_t	Amount of vehicles in period t.
z_e	Describes, if a probability adjustment is performed $z_e \in \{0, 1\}$.
τ_e	Required detection redundancy for the respective event type e.
x_e	Size of a data packet, with a detected event type e.
b_e	Size of a control message, regarding event type e.
$p_{e,t}$	Probability to transmit an event of type e in period t.
$n_{e,t}$	Total number of sensed events of type e in period t.
$F_{tot(e,t)}$	Total amount of data traffic for event type e in period t.
$F_{meas(e,t)}$	Amount of measured data traffic for event type e in period t.
$F_{adj(e,t)}$	Amount of adjusted data traffic, i. e., control messages, for event type e in period t.
$l_{max,e}$	Tolerated detection latency for event type e.
$L_{e,t}$	Set of all latencies of all events of type e, measured in period t.
$l_{e,t,\rho}$	The ρ % quantile of $L_{e,t}$ in period t.
δ_e	Desired detection density for event type e.
$\delta_{e,t}$	Measured detection density for e in period t.

of $e \in E_D$, we multiply the term by a weighting γ, to ensure a distinct adjustment in case of the quality metric is not fulfilled. In case of $e \in E_D$, the term penalizes, if the ρ % quantile of all measured latencies, regarding event type e in period t exceeds the tolerated detection latency $l_{max,e}$. In case of $e \in E_C$ the term penalizes, if the measured detection density for e in period t undercuts the desired detection density. The quadratic term converges to zero for a measured metric, approximating the desired metric. Thus, we add 1 to the term to ensure a minimum value of weighting ϵ_e of 1, because an insufficient incoming data rate is penalized, but an exceeding data rate is not rewarded.

Overall the weighting ϵ_e is determined by the deviation of the measured quality metric and the desired quality metric. The data traffic $F_{tot(e,t)}$ consists of the data traffic of transmitted events and the respective control traffic. Within this term, the transmission probability $p_{e,t}$ and the decision z_e, if a control message will be sent,

can be influenced. To give an overview about the previously introduced parameters, we summarize them in Table 5, with an according description.

$$\forall t \in T, e \in E, m \in M : \min\left(F_{tot(e,t)} \cdot \epsilon_e\right) \quad \text{with} \tag{7}$$

$$\forall e \in E_D : \epsilon_e = \begin{cases} \gamma \cdot \left(\frac{l_{e,t,\rho} - l_{max,e}}{l_{max,e}}\right)^2 + 1 & \text{if} \quad l_{e,t,\rho} > l_{max,e} \\ 1 & \text{else} \end{cases}$$

$$\forall e \in E_C : \epsilon_e = \begin{cases} \left(\frac{\delta_e - \delta_{e,t}}{\delta_e}\right)^2 + 1 & \text{if} \quad \delta_e > \delta_{e,t} \\ 1 & \text{else} \end{cases}$$

$$\forall e \in E : F_{tot(e,t)} = F_{meas(e,t)} + F_{adj(e,t)} = (n_{e,t} \cdot p_{e,t} \cdot x_e + z_e \cdot b_e \cdot va_t)$$

This formal specification is the basis for our system approach that we describe in the following. We start with a description of the backend in Section 5.2.2, followed by the cellular based client side of ProbSense.KOM, in Section 5.2.3 and the client side of Hybrid-ProbSense.KOM, extended by hybrid communication, in Section 5.2.4.

5.2.2 *Backend*

The backend side is responsible for balancing data transmissions from the vehicles, to ensure a low transmission rate and concurrently stick to certain quality metrics. Moreover, received information is stored in a database at the backend side. This balancing is done by controlling the vehicles' transmission probabilities, individually for each event type and separately for each geo cell. These transmission probabilities are calculated at the backend side and propagated towards the vehicles, within the respective geo cells. On the vehicle side, these probabilities are used as probabilistic decision model, to determine if a detected event should be transmitted. As result, the amount of transmitted data is controlled by the calculated transmission probabilities. These calculated probabilities also affect the quality metrics of data collection, i. e., the recognition latency of the received events. First of all, we consider the balancing in the following.

Balancing: Statistically, the received amount of data can be cut in half, by reducing the transmission probabilities to half. Doubling the transmission probabilities, until a maximum of $p_{e,t} = 1$, will result in the opposite. Thus, if the received data rate for an according event type is not high enough at the backend to fulfill the required quality metrics, then the transmission probabilities are increased. In case the received data rate is still not high enough for the maximum of $p_{e,t} = 1$, then the vehicle density in the respective geo cell is too low. The control task on the backend side is to balance the transmission probabilities $p_{e,t} \forall e \in E$, with the result of all quality metrics being fulfilled, i. e., the event recognition latency for discrete events and received density for continuous events. The received density of continuous events $\delta_{e,t}$ can simply be measured, in contrast to latency $l(e,t)$ of discrete events. To approximate the latency, we use the redundancy of the received event detections.

Table 6: Overview and description of the introduced parameters to describe the balancing. Since every geo cell m is considered separately, the respective parameter always belongs to the one geo cell m.

Parameter	Description
$s_{e,i,t}$	A specific instance of an event of type e in period t.
$a_{s_{e,i,t}}$	The amount of received measurements of $s_{e,i,t}$ in period t.
I_s	Set of transmit times of the respective events $s_{e,i,t}$.
$I_s(n)$	The *n-th* element in set I_s.
l_s	Detection latency of $s_{e,i,t}$.

We define $s_{e,i,t}$ as a specific instance of an event of type e and $a_{s_{e,i,t}}$ as the amount of received measurements of $s_{e,i,t}$ in period t. I_s is the set of transmit times of the respective events, which is the basis to approximate the latency l_s. Hereby, $I_s(n)$ denotes the *n-th* element in the set I_s. As we define the latency l_s as the time from event occurrence till storage in the backend database, i. e., the required redundancy τ_e is received, it can be calculated as the average between the first and the last event transmission, multiplied by the required redundancy τ_e. The assumption hereby is a relatively homogenous traffic within the respective geo cell, which in turn is a requirement for the definition of geo cell size and length of adjustment period. Based on this, we approximate the latency of a specific event l_s with Equation 8.

$$l_s = \frac{I_s(a_{s_{e,i,t}}) - I_s(1)}{a_{s_{e,i,t}} - 1} \times \tau_e \tag{8}$$

Whereby a larger $a(s_{e,i,t})$ allows a more exact approximation of l_s. However, the calculated average latency can deviate from the theoretic average latency, for the specific transmission probability. In case of the calculated average latency is higher than the theoretical average latency, this deviation is unproblematic. The result would be a too high transmission probability, with the effect of an over-fulfilled quality metric and slightly unnecessary high event data transmissions. Nevertheless, in case of the calculated average latency is lower than the theoretical average latency, respective latencies for an upcoming event cannot be assured to be lower than the maximum tolerated detection latency $l_{max,e}$. To address this issue, we set a limit of how many transmissions have to be measured for a specific event type e in period t per geo cell m, before an adjustment of the respective transmission probability $p_{e,t}$ is allowed. In addition, we divide each geo cell m into four sub cells. The latency l_s is determined separately for each of these four sub cells and only the highest value is used for further computations. An issue of a probabilistic transmission model, that should be minimized, is variation in the results. Several models, that are mostly based on past values, are available to decrease this variation. Appropriate mechanisms are the use of averaging, the moving average, or the method of least squares. In our implementation we use the latter one, as mechanism to predict the latency $l_{e,t+1}$ of the next adjustment period, since a change in transmission probabilities only affects the next adjustment period. The method of least squares is ascribed to Gauss and is, among others, described by Stigler [146]. The concept is to find a graph, that fits the measurement points in an overall optimum. This optimum is defined as the minimum of the sum of the squared distance, between the measured values and the respective

values of the graph. The formula of least squares is given in Equation 9, in which y_i are the measurement points, $f(x_i)$ the fitting function and S needs to be minimized.

$$S = \sum_{i=0}^{n} (y_i - f(x_i))^2 \tag{9}$$

The selection of an appropriate fitting function $f(x_i)$ depends on the respective scenario and measurement data. We choose a linear function, since we only want to predict the next value and expect only slight changes of the vehicle density, due to sufficient short adjustment periods t. An overview about the parameters used to describe the balancing, is given in Table 6.

Continuous Events: Continuous events do not appear at certain dates and thus, respective transmission probabilities cannot be calculated based on the latency. Instead, the respective transmission probability $p_{e,t}$ is solely dependent on the amount of incoming events per geo cell m, i. e., the event reception density. This event reception density equals the amount of received events of type e, divided by the region size of geo cell m and the time of the adjustment period t. We use the amount of events per square kilometer and hour as unit for the event reception density. The respective transmission probability $p_{e,t+1}$ is calculated by the ratio of δ_e, the desired detection density for event type e, and $\delta_{e,t}$, the measured detection density for e in period t. The new transmission probability $p_{e,t+1}$ is a linear transformation of $p_{e,t}$, with the calculated ratio, as given in Equation 10. Firstly, we approximate the number $n_{e,t}$, how often an event of type e has been sensed in period t, by division of $a_{e,t}$, the amount of received measurements of event of type e in period t, with the respective set transmission probability $p_{e,t}$. Parameters, newly introduced to describe the handling of continuous events, are summarized in Table 7.

$$p_{e,t+1} = \mathrm{Min}\left(\frac{\delta_e}{\delta_{e,t}} \cdot p_{e,t}; 1\right) \tag{10}$$

Discrete Events: The adaption of the transmission probability $p_{e,t}$ is based on measured latencies. We use the average latency as basis for calculating adjustments of transmission probabilities $p_{e,t+1}$ for the next period. This is done, because directly using the ρ% percentile has shown in simulations a high fluctuation in the received latency values. This is because of the relative low amount of values for short adjustment periods. Our goal is that 99%, or 95% respectively, of the detected events are

Table 7: Overview and description of newly introduced parameters for the description of continuous events.

Parameter	Description
$a_{e,t}$	The amount of received measurements of events of type e in period t.
$p_{e,t}$	The transmission probability $p_{e,t}$.
ϕ	The cumulative distribution function (cdf) of the negative binomial distribution.
σ	Standard deviation of the negative binomial distribution.
α	Significance level, i. e., the tolerated error of precision.

transmitted within the tolerated maximum latency $l_{max,e}$. Here, we set our tolerated error to $\alpha = 0.01$ and $\alpha = 0.05$ in our experiments, i. e., we want to achieve a precision of 99% and 95%, respectively. We calculate the average latency $l_{avg,e}$ and the latency, which is undercut by $\rho = 99\%$, or respectively $\rho = 95\%$, of the received events. However, the adjustment period has to be chosen long enough, to ensure most events are transmitted at least two times, because this is the minimum to calculate a respective latency. If an event is received only once, it is shifted into the calculation of the next period $t + 1$. In the following, we name the average time between two vehicles passing a specific event type, the *passing time* $\bar{t}_{pass,e}$. Our process on the backend side, to adapt the transmission probability for discrete events, works as follows: After each measurement period t, the system calculates the average latency $\bar{l}_{e,t}$ of all incoming events of type e, i. e., the measured average latency, as given in Equation 11.

$$\bar{l}_{e,t} = \frac{1}{a_{e,t}} \cdot \sum_{i=0}^{a_{e,t}} l_{e,t,i} \tag{11}$$

Next, we want to calculate the $(1-\alpha)\%$ percentile of the latency. Since this calculation is not directly possible, we utilize the distribution function of the negative binomial distribution. We approximate the $(1-\alpha)\%$ percentile of the latency $l_{(1-\alpha)\%}$, by the ratio of the $(1-\alpha)\%$ percentile and the average of the negative binomial distribution. The $(1-\alpha)\%$ percentile of the negative binomial distribution $NB_{(1-\alpha)\%}(p_{e,t}, \tau_e)$ is based on our current transmission probability $p_{e,t}$ and the required detection redundancy for the respective event type e. This calculation is given in Equation 12.

$$l_{(1-\alpha)\%} = \frac{NB_{(1-\alpha)\%}(p_{e,t}, \tau_e)}{\mu_{NB(p_{e,t},\tau_e)}} \cdot \bar{l}_{e,t} \tag{12}$$

We use the ratio of the $(1-\alpha)\%$ percentile $NB_{(1-\alpha)\%}(p_{e,t}, \tau_e)$ and the expectation $\mu_{NB(p_{e,t},\tau_e)}$ and multiply this ratio with our measured average latency $\bar{l}_{e,t}$, to approximate the $(1-\alpha)\%$ percentile of the measured latencies $l_{(1-\alpha)\%}$. In other words, we approximate the $(1-\alpha)\%$ percentile of the current latency $l_{(1-\alpha)\%}$, by multiplication of the measured average latency $\bar{l}_{e,t}$, with the ratio of the $(1-\alpha)\%$ percentile of the negative binomial distribution $NB_{(1-\alpha)\%}(p_{e,t}, \tau_e)$ and the expectation $\mu_{NB(p_{e,t})}$.

The respective transmission probability $p_{e,t+1}$ is calculated by the ratio of $l_{(1-\alpha)\%}$, the $(1-\alpha)\%$ percentile of the measured latencies, and $l_{max,e}$, the maximum toler-

Table 8: Overview and description of newly introduced parameters for the description of discrete events.

Parameter	Description
$NB_{(1-\alpha)\%}(p_{e,t}, \tau_e)$	The $(1-\alpha)\%$ percentile of the negative binomial distribution for probability $p_{e,t}$ and required detection redundancy τ_e.
$\bar{l}_{e,t}$	Average latency of all incoming events of type e.
$l_{e,t,i}$	Single measured instance i of latency $l_{e,t}$.
$a_{e,t}$	Amount of received measurements.
$\bar{t}_{pass,e}$	*Passing time*: Average time between two vehicles passing an event of type e.

ated latency for event type e. The new transmission probability $p_{e,t+1}$ is a linear transformation of $p_{e,t}$, with the calculated ratio of $l_{(1-\alpha)\%}$ and $l_{max,e}$, as given in Equation 13.

$$p_{e,t+1} = Min\left(\frac{l_{(1-\alpha)\%}}{l_{max,e}} \cdot p_{e,t}; 1\right) \tag{13}$$

Adjustment: In our approach, we assume a cellular connection between the vehicles and the backend. Propagation of probability adjustments would require a unicast transmission to each vehicle, if no special broadcast technology, like eMBMS, would be used. Due to these transmission costs for the probability adjustment itself, an according update should not always be propagated. Such an adjustment is only performed, if the expected savings in the measured data traffic are larger than the expected data traffic for a transmission probability update. Algorithm 1 shows our decision algorithm of a probability adjustment. In case the newly calculated probability $p_{e,t+1}$ is higher than the current probability $p_{e,t}$, an according probability adjustment is always performed, because in this case the quality metric is not fulfilled, and not performing that update would harm the data quality. In case of the

Algorithm 1 Probability adjustment algorithm.

```
newProb ←calcProb(actual, expected);
if newProb > oldProb then adjustProb();
else if newProb < oldProb then
    savedTraffic ←calcSavedTraffic(traffic, oldProb, newProb);
    if savedTraffic > adjustmentTraffic ·(1 + β) then adjustProb();
    end if
end if
```

newly calculated probability $p_{e,t+1}$ is less than the current probability $p_{e,t}$, the system checks, if the potential savings in data traffic are higher than the caused control data traffic. To prevent high frequent probability adjustments, and as a consequence thereof a high amount of control data traffic, we add a constant factor β onto the control data traffic. This factor β creates a hysteresis, i. e., a variant delayed behavior of the probability adjustments. In our experiments, we have set $\beta = 10\%$. The comparison, if a probability adjustment should be performed, is given in Equation 14. Whereby $F_{meas(e,t)}$ is the amount of measurement data traffic for event type e in period t and $F_{adj(e,t)}$ the amount of adjustment data traffic, i. e., control messages, for event type e in period t, as given in Table 5.

$$F_{adj(e,t+1)} \cdot (1+\beta) \leqslant F_{meas(e,t)} \cdot \left(1 - \frac{p_{e,t+1}}{p_{e,t}}\right) \tag{14}$$

This probability adjustment is performed periodically, with the adjustment period t. Overall the duration of t affects the desired accuracy.

5.2.3 *Client Side: ProbSense.KOM*

Vehicles are subscribed to the respective geo cell m, they are currently driving in. The assumption is a fixed grid of virtual geo cells, as mentioned before. Transmission probabilities $p_{e,t}$ are individual per event type e and separately calculated per

geo cell m, i. e., the probabilities may be different and individual for each geo cell. In case of changing the geo cell or on journey start, vehicles get information about the currently used transmission probabilities within the respective geo cell m. While being within a specific geo cell m, vehicles get constant updates of transmission probabilities, if necessary. Using this approach, the backend does not necessarily need to know the exact number of vehicles in the respective geo cells. In particular, tracking of vehicle positions is not necessary and adequately large geo cells provide a certain privacy. The backend adapts the transmission probabilities, considering the amount of received event detections. On the client side, detected events are transmitted according to the respective transmission probability $p_{e,t}$.

The overall system is used to gather location based information, by the use of vehicles that serve as mobile sensors. Gathered information consists of discrete events, e. g., the detection of traffic signs, as well as continuous events, e. g., gathering temperature values or the rain drop rate. At the backend, this information is used to provide a respective long-range information view towards participating vehicles. Thus, participating vehicles should benefit from gathered information, they also contribute to. Providing information, related to road segments in driving direction, typically based on static map information, is known as electronic Horizon (eHorizon). Based on this notion, we have introduced the concept of a remote eHorizon in our previous work [147], that we present in the following Chapter 6. Such a remote eHorizon can be described as connected service that provides information along the current driving path of a vehicle. This information can consist of relatively static components, e. g., the road network and serves as local map update or complement, transient static information, e. g., current status of traffic signs, or higher dynamic information, like traffic light status, traffic status, accidents or local weather information. A more detailed view on possible information types is given in Chapter 4, within the presentation of use case dependent information demands. In general, such a remote eHorizon provides the capabilities for long-range predictive ADASs. However, such a system also enables an optimized event detection and data upload process within systems like ProbSense.KOM. By a comparison of locally sensed information and information provided within the eHorizon, the vehicle side can decide if sensed information is relevant for transmission. Moreover, this enables the detection of absence of previously detected information, i. e., forgetting is supported. An according decision process at the vehicle side is illustrated in Figure 10. The first step if new sensed information is available, i. e., an event is detected, is to determine with probability $p(e)$, if it should be transmitted or discarded. In case of continuous events, i. e., a continuous measurement task, respective data is directly transmitted. In case of discrete events, respective data is also directly transmitted, if no eHorizon information is available. Else, respective data is only sent, if the information is new or has changed. An extension could be an event trigger for eHorizon information, that was not sensed, which would enable unlearning, i. e., that a location based information does not exist anymore. For continuous events a send on delta strategy could be an extension, i. e., transmit only if a certain delta is exceeded. But our assumption of a predefined required redundancy at the backend, that allows to compensate false sensed information, would not work with such an approach. However, availability of eHorizon knowledge and an unlearning process is out of scope within our following experimental evaluation.

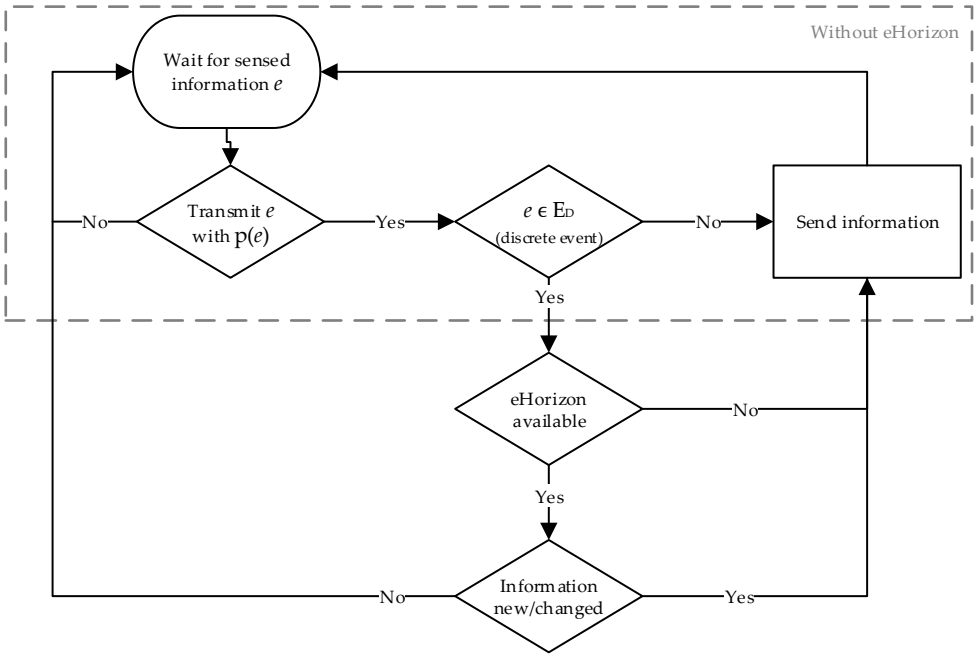

Figure 10: Overview of the client decision process, image from [96].

5.2.4 *Clint Side: Hybrid-ProbSense.KOM*

Hybrid-ProbSense.KOM is a hybrid information gathering approach, based on our previously introduced, purely cellular based approach *ProbSense.KOM*. The assumption is an increasing penetration of V2V wireless ad-hoc communication capabilities within vehicles in the next future. The hybrid extension is completely realized on the client side, i. e., it works with exactly the same backend as *ProbSense.KOM*. This brings a further advantage, that the backend dos not need detailed knowledge about the clients, which is a benefit in terms of privacy.

As the backend works unmodified, clients transmit sensed events, based on the assigned transmission probability, according to the respective geo cell. This transmission process, to the backend via cellular networks, basically works similar to our previously introduced approach *ProbSense.KOM*. However, in case of a client has also wireless ad-hoc communication available, a sensed event is always broadcasted via the V2V ad-hoc channel. Vehicles in communication range, also equipped with V2V communication capabilities, save this information, including a time stamp and sender ID. If a vehicle now senses an event and decides for a transmission, according to the assigned transmission probability, then a list of all n known detections is transmitted. Thus, at the backend, this received information can be treated as n separate sensed events. In this way, a set of n sensed events can be transmitted within one data packet. The according transmission probability can be reduced, based on the higher rate of incoming sensed information. This allows to reduce the wireless communication costs for the cellular link.

For the V2V ad-hoc communication range, we assume in the following 150m, according to our findings in Section 4.1. In our following experimental evaluation, we iterate the V2V penetration rate.

5.3 EVALUATION

In this section we present our evaluation results of *ProbSense.KOM* and *Hybrid-ProbSense.KOM*. To begin with, we start with a detailed description of our simulation configuration and setup in Section 5.3.1. Afterwards, we compare our solutions *ProbSense.KOM* and *Hybrid-ProbSense.KOM* against an opportunistic transmission model in Section 5.3.2. We show a possible reduction in the data transmission volume of up to 50% in comparison to opportunistic approaches for our cellular based approach *ProbSense.KOM*. In case of our hybrid approach *Hybrid-ProbSense.KOM*, we show a further reduction in data transmission volume of up to 15%.

5.3.1 *Evaluation Setup*

In the following, we describe the configuration and setup of our evaluation. First, we describe the used toolset in Section 5.3.1.1, followed by a description of the used simulation scenario in Section 5.3.1.2. Afterwards, we present our simulation parameters and assumptions, as well as evaluation metrics in Section 5.3.1.3.

5.3.1.1 *Simulation Environment*

We have implemented our prototypes of *ProbSense.KOM* and *Hybrid-ProbSense.KOM* in Java. The implementation consists of an implementation of the backend side and an implementation for the vehicles, each for *ProbSense.KOM* and *Hybrid-ProbSense.KOM* respectively. Sensed information is serialized into a Google Protocol Buffers[2] structure, according to the *Vehicle Sensor Data Cloud Ingestion Interface Specification*, published by HERE[3] [148]. Using this data format, we have a resulting data packet size of 123 Bytes per transmitted event, and 15 Bytes per probability adjustment in the control data packet. Our considered scenario, described in detail in Section 5.3.1.2, is a large conurbation, which typically has a very high cellular network coverage and would consist of several LTE cells. Since we are considering a system for low data rate information gathering, the used data transmissions are in the range of much less than 1% of the LTE network capacity [149]. Thus, we only add up transmitted data packets and do not simulate any network effects, i. e., we do not consider any packet loss or network congestion.

We use the open source Simulation of Urban MObility (SUMO[4]) simulator, regarding the simulation of the road network and vehicular traffic flow [150]. The tool allows to simulate traffic flow within a scenario, that is based on a map. Movement is based on a lane-changing model and a car-following model [151]. On simulation start, a seed parameter can be set as input for the Mersenne Twister pseudorandom algorithm. Generated numbers influence the traffic flow within the simulation, e. g., influence vehicle parameters, like maximum speed, speed variation or car following behavior. We have executed each simulation run configuration 30 times, each with a different initial seed parameter value. Presented results are the averaged results of the respective repetitions.

2 https://developers.google.com/protocol-buffers/

3 https://lts.cms.here.com/static-cloud-content/Company_Site/2015_06/Vehicle_Sensor_Data_Cloud_Ingestion_Interface_Specification.pdf

4 http://sumo.dlr.de/

5.3.1.2 *Simulation Scenario*

A Simulation of Urban MObility (SUMO) scenario is based on a map, that consists of nodes, that represent intersections and edges, that represent roads. Each edge is an unidirectional segment between two intersections and can consist of several lanes. All simulation description files are *XML* based.

Two large and freely available simulation scenarios exist for the cities of Cologne, named TAPAS Cologne, and Luxembourg, named Luxembourg SUMO Traffic (LuST) [152, 153]. We have used both within our simulations and describe them in more detail in the following. Both consist of different road types, including residential roads, arterial roads and highways.

TAPAS Cologne: The Cologne scenario is based on OSM map material and the *Travel and Activity Patterns Simulation (TAPAS)* model [152]. TAPAS is a system to calculate travel demands for a synthetic population, based on empirical data. To model preferably realistic vehicular traffic, a set of 30700 activity reports from about 7000 households have been used for the TAPAS Cologne scenario. The scenario consists of 0.6 million individual vehicle routes, a map size of about 32.3 $\mathrm{km} \times 34.4\ \mathrm{km}$, i. e., about 1111 km^2, around the city of Cologne and 4500km road network. In total, the TAPAS Cologne scenario consists of 24 hours simulation time, but only a subset of two hours, beginning at simulation time 6:00 a.m., is freely available. We have used this two hour subset within the following simulations. During these two hours the traffic is constantly increasing to a maximum of about 13,300 vehicles driving in parallel.

Luxembourg SUMO Traffic (LuST): The Luxembourg SUMO Traffic (*LuST*) scenario has a map size of about $13.6\ \mathrm{km} \times 11.5\ \mathrm{km}$, i. e., about 156 km^2, around the city of Luxembourg [153]. It consists of about 2365 intersections and 5959 connecting roads. The total length of the road network is about 931 km. In addition, the *LuST* scenario contains public transport, i. e., bus routes and stops. The scenario is also based on OSM map material. Vehicular travel demand is generated by the *ACTIVITYGEN*[5] framework, that synthetically generates activities, based on parameters like opening hours of shops, ratio of inhabitant to jobs, demographic information or time of schools. Available data within OSM and open government data from Luxembourg was used as input for the *LuST* scenario. In total, the *LuST* scenario consists of 24 hours simulation time, that we have completely used within the following simulations. During these 24 hours, altogether 295,979 vehicles are inserted into the scenario. The traffic density is very low during the night and it is increasing as from about 4:00 a.m., with three peaks at 8:00 a.m., 1:00 p.m and 7:00 p.m., and a maximum of about 6000 vehicles driving in parallel.

5.3.1.3 *Simulation Parameters & Evaluation Metrics*

In our simulations we evaluate *ProbSense.KOM* and *Hybrid-ProbSense.KOM* against each other and against an opportunistic transmission model. In the opportunistic transmission model each sensed event is directly transmitted to the backend, which ensures the fastest possible event detection on the backend. We perform all simula-

5 http://sumo.dlr.de/wiki/ACTIVITYGEN

Table 9: Simulation parameters

Parameter	**Value**
Scenario	Cologne, Luxembourg
Transmission mechanism	opportunistic, ProbSense, HybridSense
Sensing density	5, 10, 20 $[\frac{measurements}{km^2 \cdot h}]$
Detection latency	5, 10, 20 $[min]$
Desired accuracy	99%, 95%
V2V penetration rate	25%, 50%, 75%, 100%

tions for the previously mentioned Cologne and Luxembourg scenarios. Regarding the vehicular traffic density, we consider the standard traffic density, preconfigured within the respective scenarios. Regarding continuous events, we consider a desired sensing density of 5, 10 and 20 $measurements$ per km^2 and $hour$. Regarding discrete events, we consider a desired detection latency of 2, 5 and 10 minutes. With respect to the desired accuracy, we consider a tolerated error α of 1% and 5%, i. e., a desired accuracy of 99% and 95% respectively. We have set the required redundancy τ to a fixed value of three. For cellular communication, we assume a penetration rate of 100% for all vehicles under consideration. With regard to our hybrid communication approach, we iterate the V2V penetration rate over 25%, 50%, 75% and 100%. For both scenarios, we randomly place 100 event locations along the 95% mostly used road segments, that trigger a discrete event on passing vehicles. In a second event placement strategy, we placed 1000 event location completely random throughout the entire scenario. For both event type classes, discrete and continuos, we consider one abstract single type. We have used different geo cell patterns in our simulations, to evaluate the influence of the geo cell size. For this, we divide the map into equally sized rectangles, since the shape has no direct influence, but the vehicular density and homogeneity. We consider the whole scenario as one single geo cell and in addition we divided the scenario into 2x2, 4x4, 8x8, 16x16 and in case of Cologne also 32x32 geo cells. This results in a geo cell size of down to about 1km edge length, in particular 1042 m in case of the Cologne scenario and 781 m in case of the Luxembourg scenario. Table 9 gives an overview about our simulation parameters.

5.3.2 *Evaluation Results and Discussion*

In this section, we evaluate the performance of our system of probabilistic gathering vehicular sensed data. To begin with, we compare *ProbSense.KOM* with the opportunistic transmission model for discrete events. We used the scenario of Cologne with a desired detection latency of 10 minutes and different geo cell sizes. The results are depicted in Figures 11, 12, 13 and 14. Confidence intervals were computed using Student's t-distribution [154]. We provide the respective evaluation result data in Tables 16, 17, 18 and 19 in Section A.3.1. We have used two different event deployment strategies. In the first setup, we placed 100 event locations randomly, along the road segments with the highest 95% traffic density. This means, we have neglected the road segments with the lowest 5% traffic density for the event placement. As met-

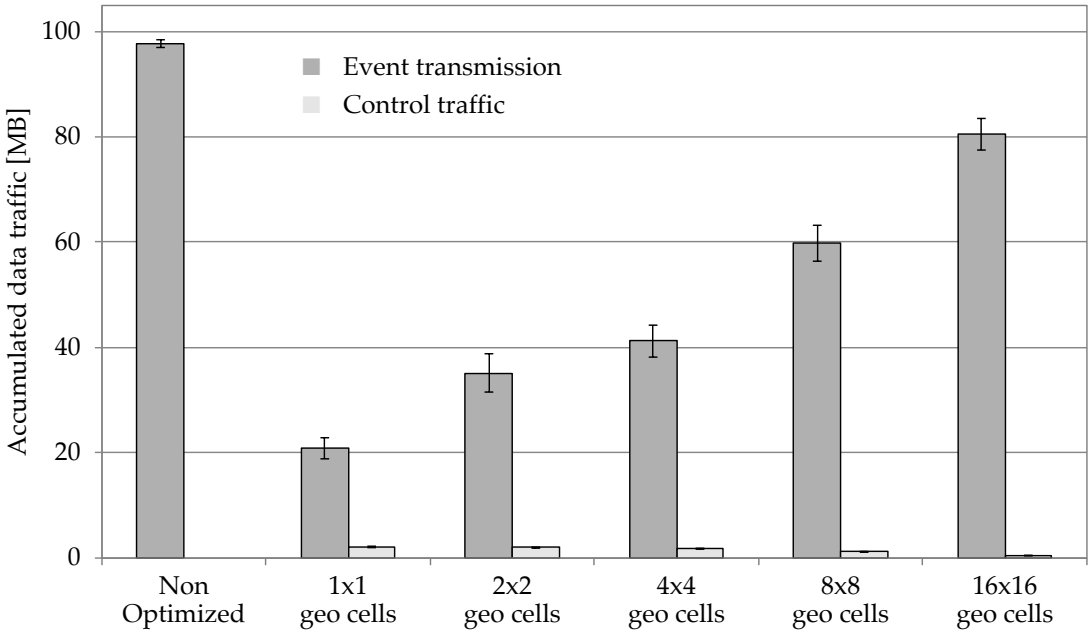

Figure 11: Comparison of *ProbSense.KOM* with the opportunistic transmission model for discrete events. The figure shows the results for the scenario of Cologne, with a tolerated error of $\alpha = 5\%$ and a detection latency of 10 minutes.

ric, we compare the accumulated data traffic throughout the whole scenario. Figure 11 gives the results for a desired accuracy of 95%, i. e., $\alpha = 5\%$, Figure 12 the respective results for a desired accuracy of 99%, i. e., $\alpha = 1\%$. The different values of α have only a minor effect. The degree of accuracy is very high in both cases, but it can be seen that a decreased accuracy allows to reduce the amount of transmitted data.

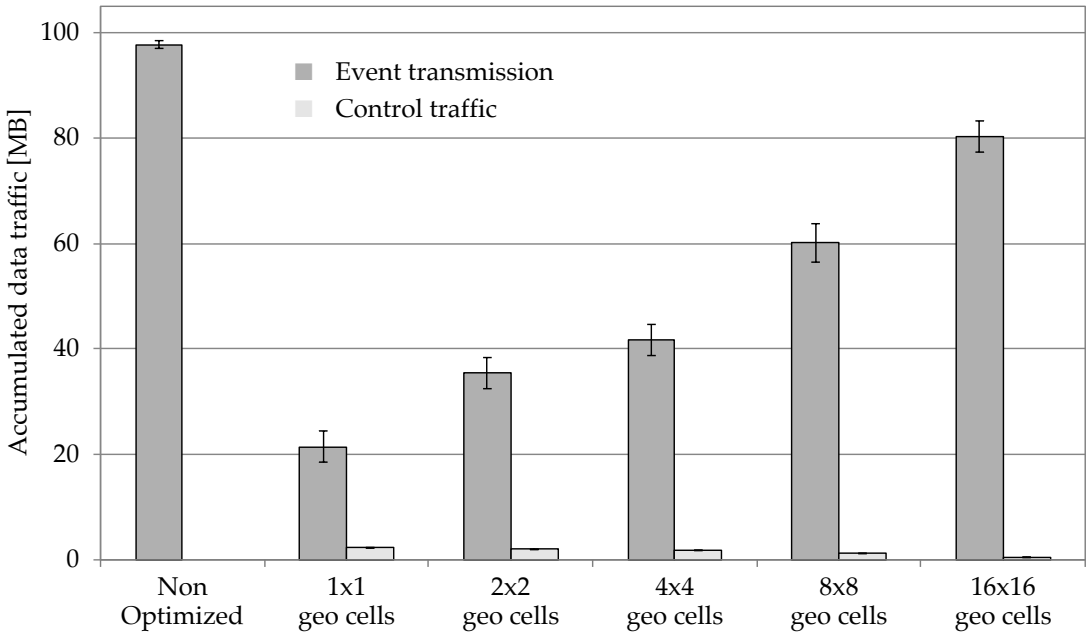

Figure 12: Comparison of *ProbSense.KOM* with the opportunistic transmission model for discrete events. The figure shows the results for the scenario of Cologne, with a tolerated error of $\alpha = 1\%$ and a detection latency of 10 minutes.

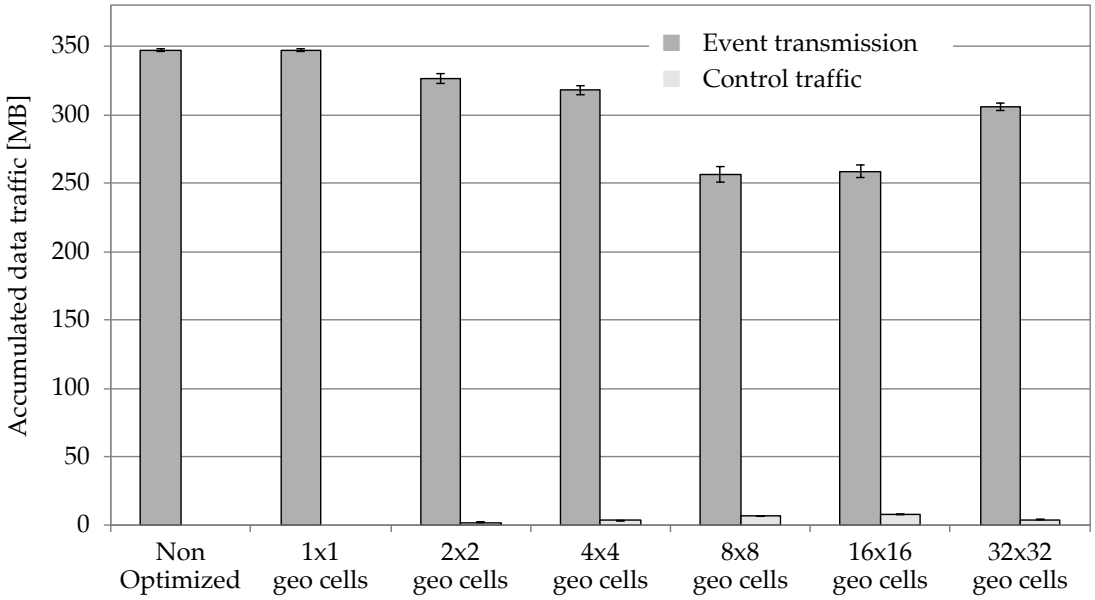

Figure 13: Comparison of *ProbSense.KOM* with the opportunistic transmission model for discrete events. The figure shows the results for the scenario of Cologne, with a tolerated error of $\alpha = 5\%$ and a detection latency of 10 minutes. In this example events are completely randomly placed throughout the scenario.

The respective accumulated data traffic in case of $\alpha = 1\%$, i. e., Figure 12, is less than 4% increased for all run configurations, compared to the case of $\alpha = 5\%$, i. e., Figure 11. The *ProbSense.KOM* approach can significantly reduce the amount of transmitted data compared to the opportunistic transmission model. In case of considering the scenario as only one geo cell, the approach shows a reduction of about 79%. However, the results show an unexpected increase in the amount of transmitted data for an increased number of geo cells, i. e., a decreased geo cell size. The explanation for this is the biased distribution of the events in combination with a relatively low number of events. This causes many geo cells to be not optimizable, because of a high fluctuation of the measured latency. This is caused by the low number of events in the respective geo cells. Since all events are placed along the highly trafficked road segments, the optimization works best for considering the whole scenario, i. e., all event locations are passed by a roughly equal amount of vehicles. This also causes the large reduction in the total data traffic, because of the homogeneity of the traffic along the event locations. To proof this, we also used another event deployment strategy.

Influence of another event deployment strategy: In the second event deployment strategy, we placed 1000 event locations completely randomly throughout the entire scenario. This shows a completely different result. Figure 13 gives the results for a desired accuracy of 95%, i. e., $\alpha = 5\%$, Figure 14 the respective results for a desired accuracy of 99%, i. e., $\alpha = 1\%$. The different values of α have also only a minor effect. The respective accumulated data traffic in case of $\alpha = 1\%$, i. e., Figure 14, is less than 5% increased for all run configurations, compared to the case of $\alpha = 5\%$, i. e., Figure 13. In both cases, degree of accuracy is very high, but it can be seen that a decreased accuracy allows to reduce the amount of transmitted data. In case on a more signifi-

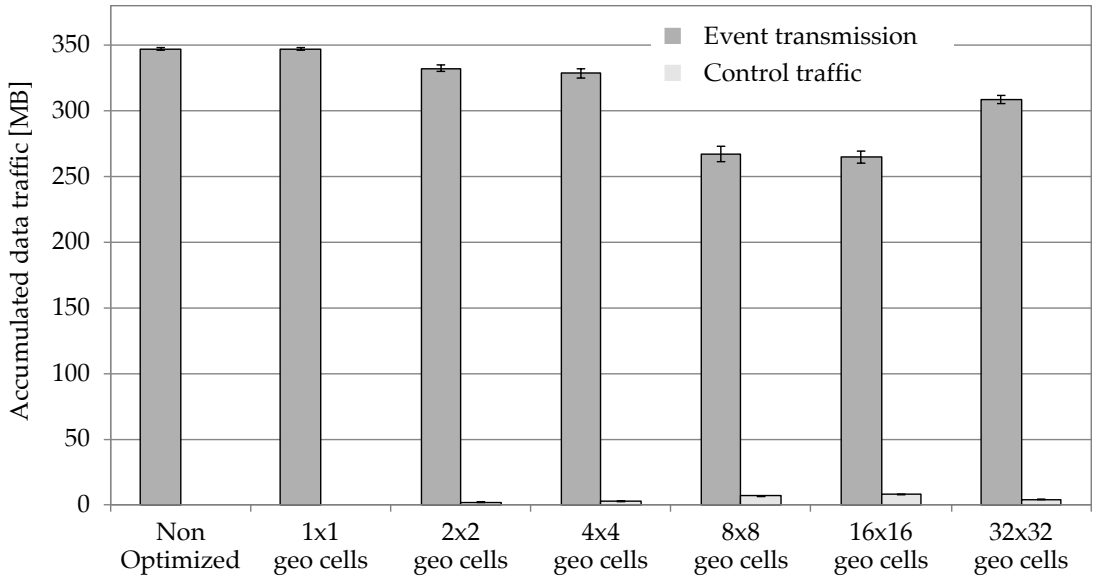

Figure 14: Comparison of *ProbSense.KOM* with the opportunistic transmission model for discrete events. The figure shows the results for the scenario of Cologne, with a tolerated error of $\alpha = 1\%$ and a detection latency of 10 minutes. In this example events are completely randomly placed throughout the scenario.

cantly decreased accuracy, the resulting amount of transmitted data might also drop more significantly. The *ProbSense.KOM* approach can still considerably reduce the amount of transmitted data, compared to the opportunistic transmission model, but only with a maximum reduction of about 26.3%, as given for 8x8 geo cells. No optimization is possible in case of considering the whole scenario as only one geo cell. This is because of a high traffic inhomogeneity, with respect to the event locations. Thus, many events are not passed by a sufficient amount of vehicles, which results in high measured latencies. This increases the average measured latency, which reduces the optimization potential. In case of a decreasing geo cell size, the optimization potential increases. The setting of 8x8 geo cells, i. e., a geo cell size of about 4 km x 4 km, shows the largest reduction in the total data traffic. The results for 8 and 16 are nearly comparable. For the further decreased geo cell size, i. e., 32x32 geo cells, we can see an increase of the data traffic. In case of a too small geo cell size, the previously mentioned traffic inhomogeneity and fluctuation comes into effect. Due to the required redundancy, too small geo cells allow no adjustment of the transmission probability, because of an insufficient average amount of vehicles passing the single event locations. Overall, we can see an increase of the necessary control traffic in case of a stronger decreased total data traffic. This is due to the higher amount of an adaptation of the transmission probabilities. However, the amount of control traffic is very low, compared with the potential savings in the overall data traffic.

Influence of the desired detection latency: Next, we consider the influence of the desired detection latency. Figure 15 shows a comparison of *ProbSense.KOM* with the opportunistic transmission model for discrete events and for a detection latency of 5, 10 and 20 minutes. The figure shows the results for the scenario of Cologne, with a tolerated error of $\alpha = 5\%$ and $\alpha = 1\%$, for the configuration of 8x8 geo cells. The event placement corresponds to the first setup, i. e., we placed 100 event locations

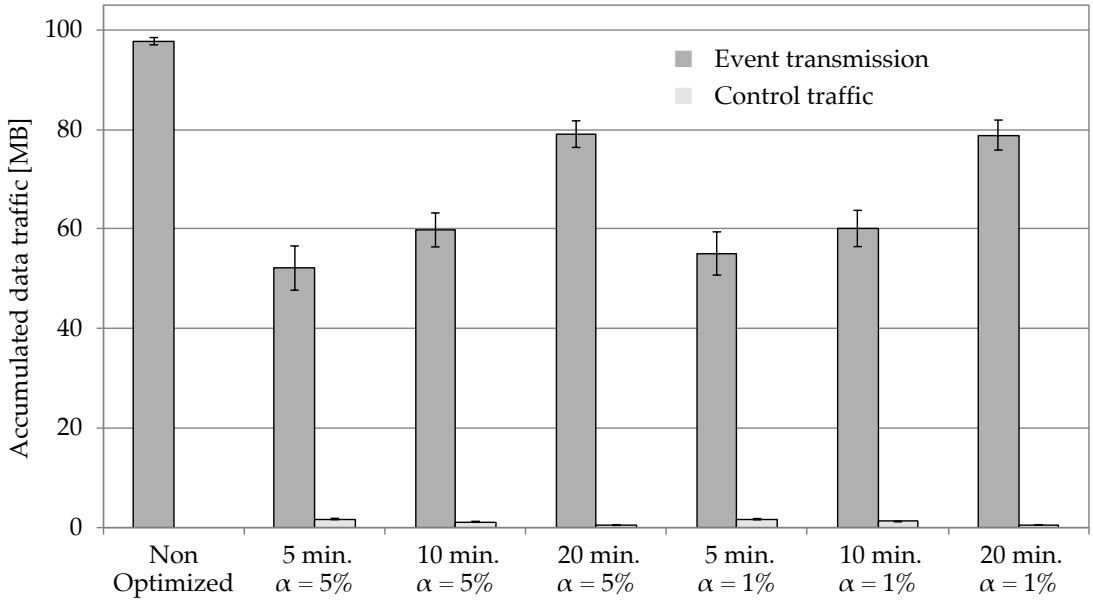

Figure 15: Comparison of *ProbSense.KOM* with the opportunistic transmission model for discrete events. The figure shows the results for the scenario of Cologne, with a tolerated error of $\alpha = 5\%$ and $\alpha = 1\%$, for a detection latency of 5, 10 and 20 minutes and 8x8 geo cells. In this example, we dropped the road segments with the lowest 5% traffic density for event placement.

randomly along the road segments with the highest 95% traffic density. We provide the respective evaluation result data in Table 20 in Section A.3.1. Also here, the different values of α have only a minor effect. The respective accumulated data traffic in case of $\alpha = 1\%$ is less than 6% increased for a detection latency of 5 minutes, compared to the case of $\alpha = 5\%$. In case of a detection latency of 10 and 20 minutes, the effect of the different values of α is less than 1%. The effect of an increased amount of transmitted data in case of an increased detection latency is at first unexpected. The explanation is a fast variation of the traffic density in relation to the observation interval. In case of a desired detection latency of 20 minutes, the adjustment period is also 20 minutes. The considered scenario has a total duration of two hours with a constant increasing amount of vehicles. This results in a constant increasing traffic density after each adjustment of the transmission probabilities. An improvement could be a sliding window approach with a shorter adjustment period or just the use of the short adjustment period of 5 minutes.

Influence of the hybrid communication approach: In the next step, we compare our probabilistic data gathering approach in case of the availability of hybrid communication, i. e., cellular communication and direct V2V communication. We iterate the V2V penetration rate from 0%, which results in the classic approach of *ProbSense.KOM*, in 25% steps towards a 100% equipment rate. We consider the scenario of Cologne, with a tolerated error of $\alpha = 5\%$, for a detection latency 10 minutes and 8x8 geo cells. The event placement corresponds to the first setup, i. e., we placed 100 event locations randomly along the road segments with the highest 95% traffic density. The respective results are depicted in Figure Figure 16. We provide the respective evaluation result data in Table 21 in Section A.3.1. In this example, we are

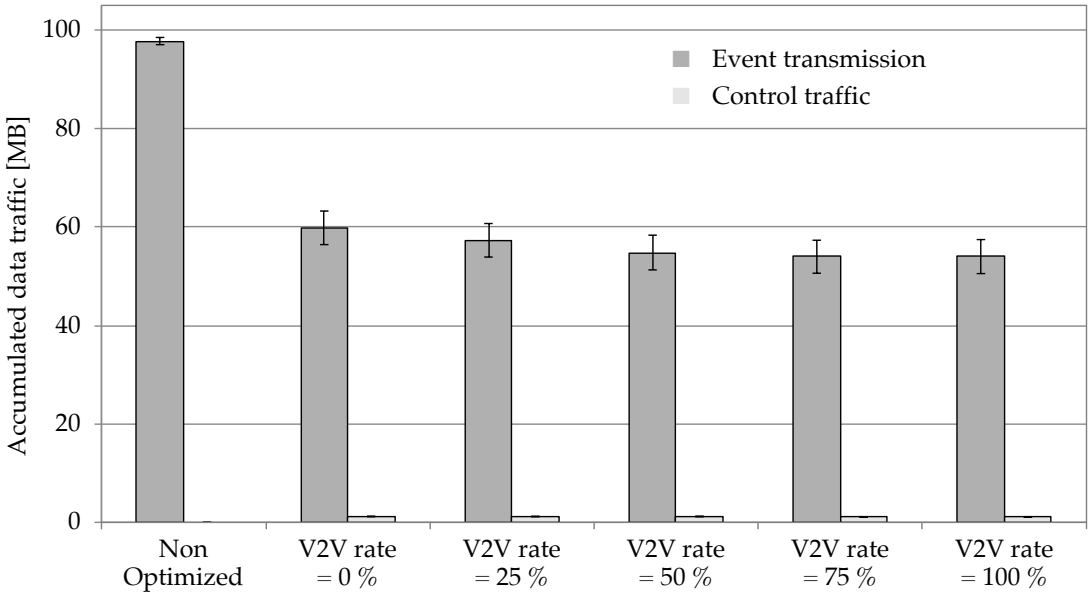

Figure 16: Comparison of different V2V penetration rates in *Hybrid-ProbSense.KOM* with the opportunistic transmission model for discrete events. The figure shows the results for the scenario of Cologne, with a tolerated error of $\alpha = 5\%$, for a detection latency 10 minutes and 8x8 geo cells. In this example, we dropped the road segments with the lowest 5% traffic density for event placement.

able to reduce the overall data traffic in case of *ProbSense.KOM*, i. e., a V2V penetration rate of 0%, by about 40%, compared to the opportunistic transmission approach. In case of V2V being available, we can see a further drop in the accumulated data traffic of about 3% in case of a penetration rate of 25%. Another 2.5% data traffic can be reduced in case of of a penetration rate of 50%, which is already close to the maximum savings. The lowest accumulated data rate, of about 55% compared to the opportunistic transmission approach, can be achieved in case of a V2V penetration rate of 100%. The results show, that the use of a hybrid communication approach is able to reduce the overall data traffic. Moreover, a V2V penetration rate of 100% is not necessary to achieve roughly the maximum savings in the data traffic.

Influence of the traffic density: It strongly depends on the traffic density, if our *ProbSense.KOM* approach is able to adjust the transmission probabilities, in order to reduce the overall data traffic. In case of a high traffic density, the opportunistic transmission approach would result in a high redundancy within the transmitted data. In case of a very low traffic density, the transmission probability can not be reduced, since the desired minimum redundancy is not satisfied. This effect is shown in Figure 17, that depicts a comparison of the opportunistic transmission model, *ProbSense.KOM* and *Hybrid-ProbSense.KOM* at three different traffic densities. The figure shows results from exemplarily selected geo cells within the scenario of Cologne, with a tolerated error of $\alpha = 5\%$, for a desired detection latency of 10 minutes and 8x8 geo cells. The event placement corresponds to the first setup, i. e., we placed 100 event locations randomly along the road segments with the highest 95% traffic density. We have selected the geo cell with the lowest, the median and the highest traffic density. In addition to Figure 17, we provide the respective evaluation result data in

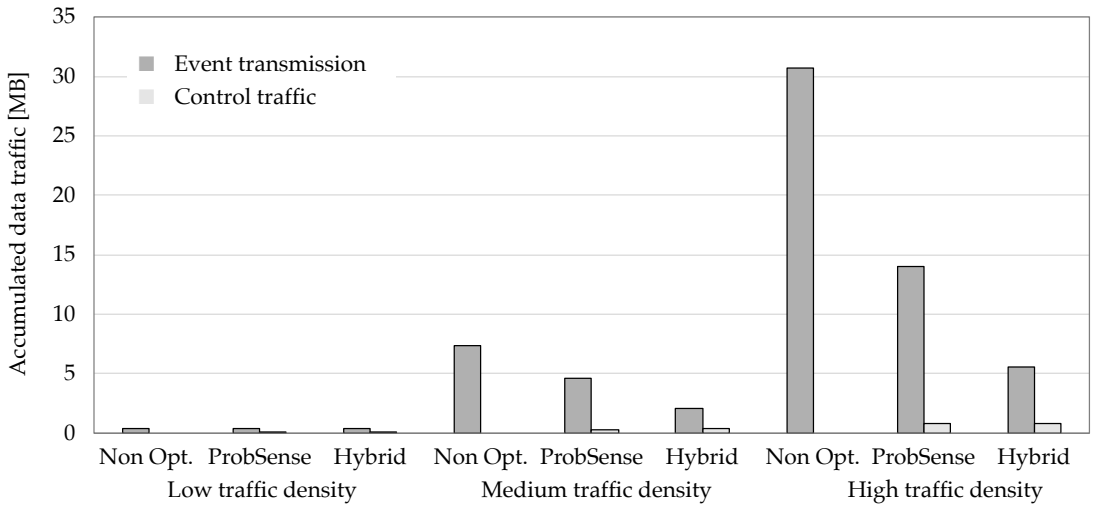

Figure 17: Comparison of the opportunistic transmission model, *ProbSense.KOM* and *Hybrid-ProbSense.KOM* at different traffic densities. The figure shows the results for exemplarily selected geo cells in the scenario of Cologne, with a tolerated error of $\alpha = 5\%$, for a desired detection latency of 10 minutes and 8x8 geo cells. In this example, we dropped the road segments with the lowest 5% traffic density for event placement.

Table 22 in Section A.3.1. In case of of a low traffic density, no optimization is possible. The medium traffic density geo cell shows a reduction of about 39% in case of *ProbSense.KOM* and a reduction of even 71% in case of *Hybrid-ProbSense.KOM*. In case of a very high traffic density, the overall data traffic significantly increases, which brings the potential for the most significant relative reduction. In this case, *ProbSense.KOM* shows a reduction in data traffic of about 55% and *Hybrid-ProbSense.KOM* shows even a reduction in data traffic of about 82%. Due to the continuously changing traffic, a particular geo cell is also varying in traffic density over time, which causes the need of a periodic adjustment of the transmission probabilities.

Compliance in accuracy: Our proposed approach, *ProbSense.KOM*, adjusts the transmission probability within a region, based on measured latencies, to reduce the overall amount of transmitted data. Concurrently, a certain maximum desired detection latency should be guaranteed, as long as a sufficiently high traffic density allows enough event detections. Since typically not all events can be passed by a sufficiently high amount of vehicles, we accept a tolerated error α, that indicates the amount of events that are permitted to be detected with a higher latency. To show the compliance of α, we consider an example in Figure 18. The depicted results are of an exemplarily selected geo cell in the scenario of Cologne, with a tolerated error of $\alpha = 5\%$, a tolerated detection latency of 10 minutes and 8x8 geo cells. The event placement corresponds to the first setup, i. e., we placed 100 event locations randomly along the road segments with the highest 95% traffic density. The figure shows the number of measured latencies per time, as relation of *ProbSense.KOM* to the non optimized approach. In other words, the figure shows the cumulative distribution of the measured latencies of *ProbSense.KOM*, in relation to the measured latencies of the non optimized approach. The dashed red lines mark the tolerated detection latency of 10 minutes and the amount of 95% of the measured latencies. It can be seen, that more

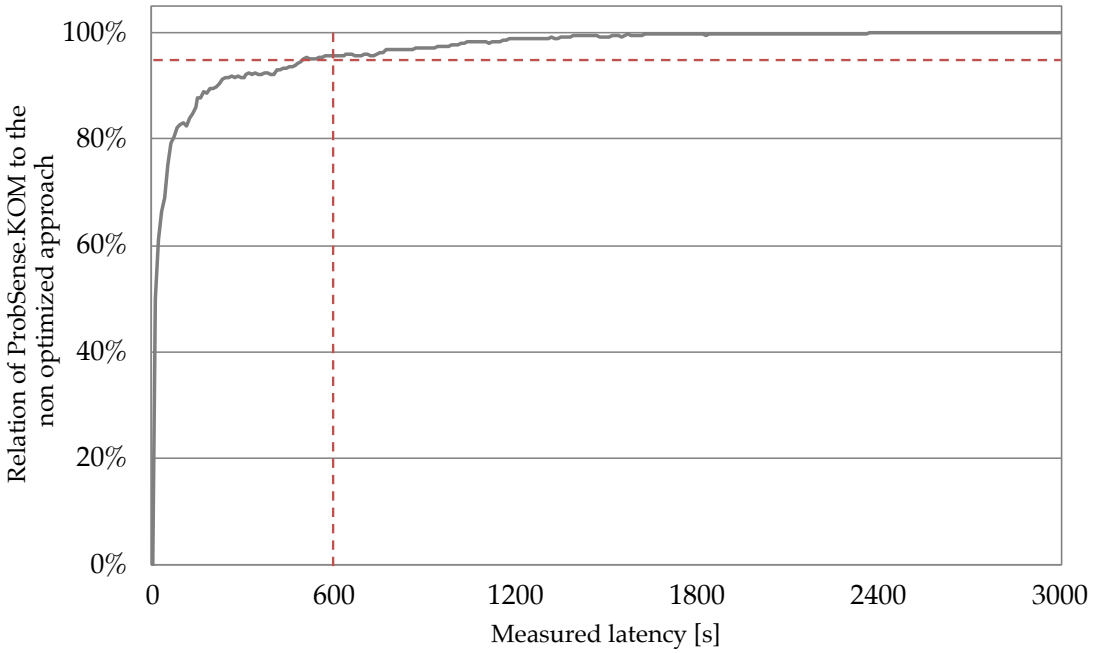

Figure 18: Number of measured latencies per time, as relation of *ProbSense.KOM* to the non optimized approach. The figure shows the result for an exemplarily selected geo cell in the scenario of Cologne, with a tolerated error of $\alpha = 5\%$, a tolerated detection latency of 10 minutes and 8x8 geo cells. In this example, we dropped the road segments with the lowest 5% traffic density for event placement. The dashed red lines mark the tolerated detection latency of 10 minutes and the amount of 95% of the measured latencies.

than 95% of the measured detection latencies are below the threshold of 600 s, i. e., 10 minutes.

Continuous events: Next, we consider the performance of *ProbSense.KOM* with respect to continuous events, for different geo cell sizes. Figure 19 gives a comparison of *ProbSense.KOM* with the opportunistic transmission model for continuous events. The figure shows the results for the scenario of Cologne for continuous events, with a sensing density of 20 $[\frac{meas.}{km^2 \cdot h}]$. In addition to Figure 19, we provide the respective evaluation result data in Table 23 in Section A.3.1. In case of the opportunistic approach, the accumulated amount of data traffic only depends on the preconfigured sensing rate and the amount of vehicles. We have set this sensing rate to one minute for all vehicles. Since the desired sensing density is only 20 $[\frac{meas.}{km^2 \cdot h}]$, the *ProbSense.KOM* approach is able to significantly reduce the amount of data traffic. In case of a smaller cell size, i. e., a higher total amount of cells, the amount of data traffic is increasing, because the system adjusts the transmission probabilities of each geo cell individually to achieve the desired detection density. Moreover, we can see from Figure 19 an increasing amount of control data traffic in case of smaller cell size, i. e., a higher total amount of cells, because of a cell change of vehicles causes control traffic for propagating the respective transmission probabilities. Thus, a large cell size would be beneficial to achieve a preferably low amount of transmitted data. However, in case of continuous events, the geo cell size might be determined by the respective application needs.

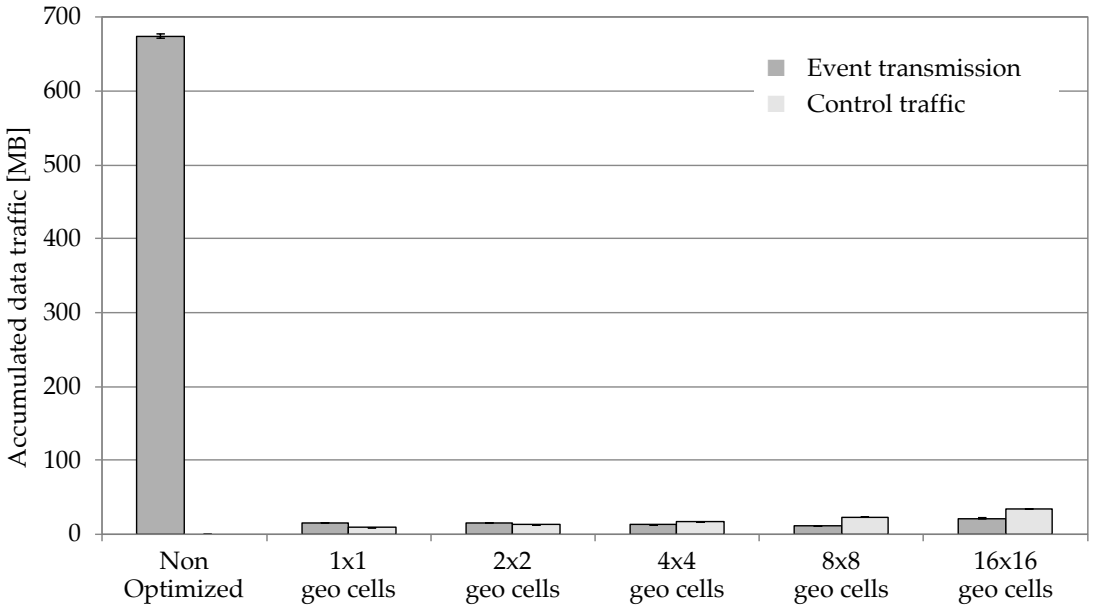

Figure 19: Comparison of *ProbSense.KOM* with the opportunistic transmission model for continuous events. The figure shows the results for the scenario of Cologne for continuous events, with a sensing density of 20 $[\frac{meas.}{km^2 \cdot h}]$. In this example, we dropped the road segments with the lowest 5% traffic density for event placement.

The following Figure 20, shows the respective results for the 8x8 geo cells setting with a variation of the desired detection density. In addition to Figure 20, we provide the respective evaluation result data in Table 24 in Section A.3.1. As expected, we can see a linear dependency of the desired detection density and the resulting amount

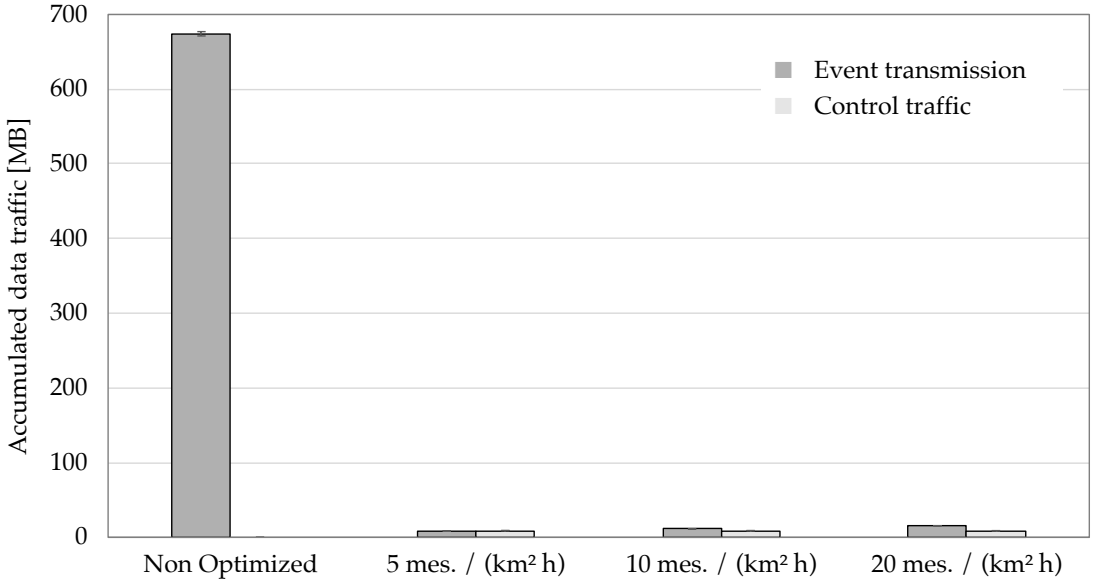

Figure 20: Comparison of *ProbSense.KOM* with the opportunistic transmission model for continuous events. The figure shows the results for the scenario of Cologne for continuous events for different sensing densities of 5, 10 and 20 $[\frac{meas.}{km^2 \cdot h}]$, in the setting of 8x8 geo cells. In this example, we dropped the road segments with the lowest 5% traffic density for event placement.

of data traffic. However, a halved desired detection density does not half the total amount of data traffic.

Results in the Luxembourg scenario: Finally, we consider our probabilistic data gathering approach in the Luxembourg scenario, to show that the previously presented results not depend on a specific scenario. In Figure 21, we give a comparison of *ProbSense.KOM* and *Hybrid-ProbSense.KOM* with the opportunistic transmission model for discrete events. The figure shows the results for the scenario of Luxembourg with a tolerated error of $\alpha = 5\%$ and a tolerated detection latency of 10 minutes, in the setting of 8x8 geo cells. Since the area of the scenario is significantly smaller than in the Cologne scenario, a setting of 8x8 geo cells corresponds, with respect to the cell size, to 16x16 geo cells in the Cologne scenario. The resulting cell size is of about 2km edge length. The V2V penetration rate in case of *Hybrid-ProbSense.KOM* is set to 100%. The event placement corresponds to the second setup, i. e., we placed 1000 event locations completely randomly throughout the scenario. This results in an about seven times higher event density, compared to the same amount of events in the Cologne scenario. In addition to Figure 21, we provide the respective evaluation result data in Table 25 in Section A.3.1. The results show an overall reduction in the accumulated data traffic, compared to the opportunistic transmission model, of about 28% in case of *ProbSense.KOM* and of about 77% in case of *Hybrid-ProbSense.KOM*. This is directly comparable to our results in the Cologne scenario for a high traffic density. A difference compared to the cologne scenario can be seen in the control traffic, that increases in case of *Hybrid-ProbSense.KOM*. This is due to higher fluctuations in the traffic density, that cause more adjustments of the transmission probabilities.

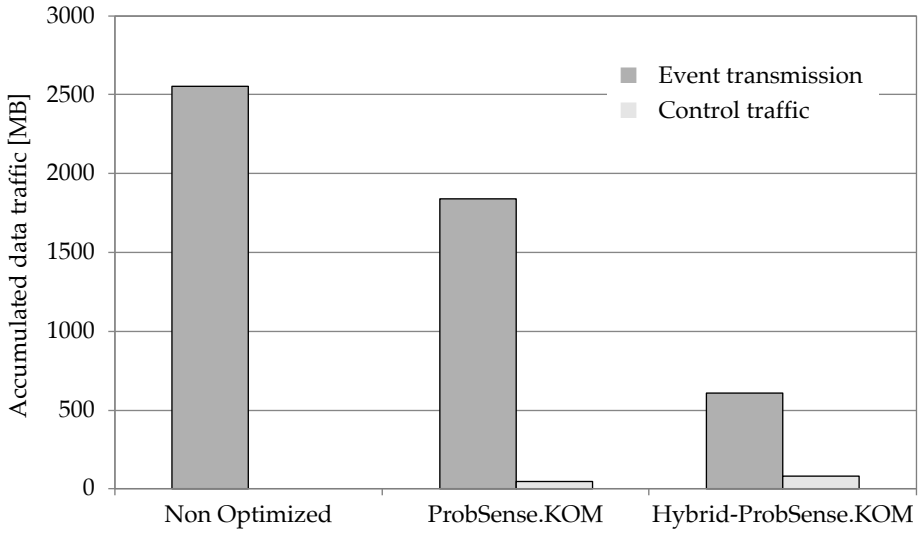

Figure 21: Comparison of *ProbSense.KOM* and *Hybrid-ProbSense.KOM* with the opportunistic transmission model for discrete events. The figure shows the results for the scenario of Luxembourg for discrete events with a tolerated error of $\alpha = 5\%$ and a tolerated detection latency of 10 minutes, in the setting of 8x8 geo cells. The V2V penetration rate in case of *Hybrid-ProbSense.KOM* is set to 100%. In this example events are completely randomly placed throughout the scenario.

5.3.2.1 *Discussion*

The provision of long-range perception information for future ADASs, requires an up-to-date data source. Since modern vehicles are equipped with a variety of sensors, these can serve as mobile sensors to gather location based information. However, with regard to the *information zone*, as introduced in Chapter 4.4, the required distance for information is above single hop inter-vehicle ad-hoc communication range (c.f. Chapter 4.1). Since a required forwarding would fail in situations of sparse traffic and at insufficient V2V penetration rates, an infrastructure support would be necessary. We consider a cellular based approach, since an extensive expansion of RSUs is very unlikely in the near future. A logically central and up-to-date information base could provide consistent information to all participants. However, the number of potentially connected vehicles to such a system requires the need of an intelligent data collection management, that is able to minimize data traffic and thus, minimize transmission costs. Concurrently, a certain maximum desired detection latency should be guaranteed, since data quality depends on the latency it takes to transmit a certain event to the backend with a defined redundancy.

Classic data collection approaches use mechanisms like local pre-processing or clustering with aggregation, to reduce the data traffic. But due to the potentially high number of connected vehicles, such a complete transmission model would cause a high redundancy in the transmitted data. As possible solution, we have introduced our probabilistic data collection approach *ProbSense.KOM*. If vehicles sense information relevant for transmission, the amount of transmitted data is controlled by transmission probabilities. A backend calculates, based on the received data rate, the appropriate transmission probability to minimize the total amount of data traffic and concurrently stick to a certain maximum desired detection latency. The adjustment of the transmission probabilities is calculated individually for each event type and separately for each geo cell. A certain maximum desired detection latency is guaranteed, as long as a sufficiently high vehicular traffic density allows enough event detections. Due to the continuously varying traffic density over time, *ProbSense.KOM* periodically adjusts the transmission probabilities. In addition, the hybrid communication extension of *Hybrid-ProbSense.KOM* allows to further decrease the total amount of cellular based data traffic, in case of the availability of V2V communication. As result, the backend can serve as an always up-to-date information source as basis to provide a long-range vehicular perception.

We have evaluated our approach with the *TAPAS Cologne* and *Luxembourg SUMO* traffic scenarios. The results have shown a possible reduction in the total cellular data traffic of about 55%, in case of the use of our probabilistic data gathering approach *ProbSense.KOM*, compared to an opportunistic transmission model. The use of our hybrid communication model *Hybrid-ProbSense.KOM*, even shows a possible reduction in the total cellular data traffic of about 82%. Moreover, our system does not require an 100% V2V penetration rate. Even a V2V equipment rate of about 25% shows significant reductions in the amount of transmitted data and a rate of about 50% is enough to achieve roughly the maximum possible reduction of cellularly transmitted data. However, the possible reduction of transmitted data strongly depends on the event density and in particular on the traffic density. In case of a sparse traffic density, no optimization is possible. This is due to the assumed required detection redundancy and maximum tolerated detection latency. Once the received data rate

at the backend exceeds the minimum required quality metric, the system starts to adjust the transmission probabilities. We have shown in our evaluation, that the system is able to stick to the required quality metrics within a predefined tolerated error. Overall, our proposed system is appropriate to efficiently collect vehicular sensed information, with respect to the total amount of cellular data traffic, and concurrently guarantees a certain data quality.

6 THE PROVISION OF THE EHORIZON AS A CLOUD SERVICE

Advanced Driver Assistance Systems (ADASs) intend to increase traffic efficiency and driving comfort, economy and safety. For the realization of such systems, one necessary requirement is detailed information about the ego vehicle status and its surroundings. This vehicular perception is realized with a set of built-in sensors [5]. But due to the physically limited and relatively short sensing range, the capabilities of ADASs are limited [6]. The concept of the electronic Horizon (eHorizon) has shown to be able to improve or even enable completely new ADAS applications. The general concept of the eHorizon has been introduced in Chapter 2.3 and is depicted in Figure 22. The illustrated eHorizon refers to the dark grey vehicle at the bottom of the picture. The green lines indicate the road geometry, that is element of the respective eHorizon. The depicted eHorizon also shows some additional information, that is related to the road geometry, i. e., traffic signs, marked by a symbol and a green position indicator. All attached information is described relative to the respective road segment.

However, existing solutions are based on relatively static map data. A *eHorizon provider* within the ego vehicle makes use of a local map database, to provide an eHorizon towards map-based ADASs [71]. Local map data has to be updated frequently, to provide a correct eHorizon and hence, reliable ADASs [123]. It is obvious, that even daily map updates would not be sufficient to also integrate more dynamic information, e. g., traffic light status. A solution is the use of a cloud service as the information provider, at least partially. However, known eHorizon concepts strictly rely on local map -based eHorizon providers. One single except is a cloud-based eHorizon presented by *Continental AG*, but no details are known, except a press release on the company's web page[1]. However, this clearly shows the significance of an up-to-date eHorizon for future ADAS and autonomous driving systems [128]. A cloud-based eHorizon could serve as logically central, always up-to-date information source. Besides other cloud services, vehicles can be used as mobile sensors to keep the cloud-based eHorizon up-to-date. A respective information gathering approach has been introduced in Chapter 5.

Our overall aim is the extension of the vehicular perception range. With respect to *information zone* related use cases (c.f. Chapter 4.4), we see the concept of an eHorizon as an appropriate mechanism for data provision. To guarantee always up-to-date eHorizon information, we propose a cloud-based eHorizon service. However, to realize such a service, several components are required, including an architecture for dynamic data management and data exchange.

An indispensable pre-requirement to create an eHorizon is a digital map database. For this, we have used Open Street Map (OSM)[2] in our implementations, since other digital maps are commercial and not freely available.

1 http://www.continental-automotive.com/www/automotive_de_en/themes/passenger_cars/interior/connectivity/pi_ehorizon_en.html

2 http://www.openstreetmap.org

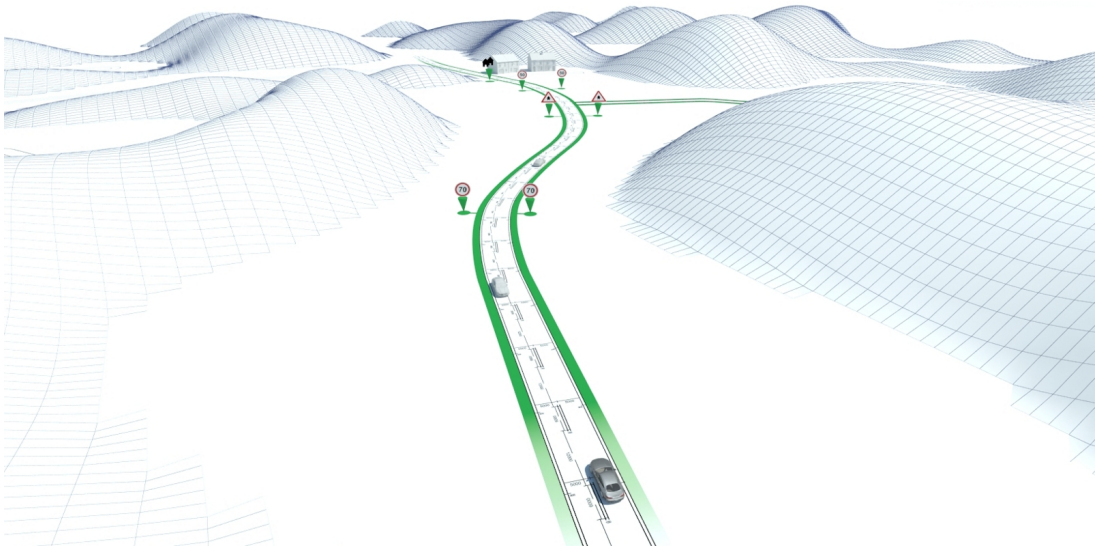

Figure 22: Illustration of the eHorizon, that contains information about road geometry and other location-based information along the driving path.

In a first step, we develop a local eHorizon provider, named *Horizon.KOM*. It is realized as local service for mobile devices and covers the basic data management structure for the eHorizon management. We have presented *Horizon.KOM* in [155]. We present a revised and extended version of the architecture of our local eHorizon provider *Horizon.KOM* in the next Section 6.1.

This basic concept has been transformed into a backend service, to serve as cloud-based eHorizon provider. We name this approach *RemoteHorizon.KOM*, as presented in [147]. In the following, we present in Section 6.2 a revised and extended version of the description of *RemoteHorizon.KOM*. Moreover, we have developed a data structure for the information exchange between the backend and the vehicles, e. g., transmitting the eHorizon. This data structure is presented in [147], [130] and in Appendix A.2.

To provide only the relevant information, it is essential to predict the most probable driving path (MPP). As mentioned in Chapter 3.4, this can be derived in most cases out of navigation system data or determined based on historical driving information. However, for the seldom case of driving an unusual route without the use of a navigation system, a heuristic is required to estimate the MPP. We developed an according heuristic based on road classes and the turn angle, that we present in Section 6.3. We have evaluated the approach in a simulation, by the use of a large scale traffic scenario with realistic traffic patterns.

The most significant proportion in the data size on an eHorizon is the description of the road geometry. This is typically described as a list of waypoints, with linear interpolation between these points, i. e., the use of a first degree polynomial. In our *RemoteHorizon.com* approach, the transmission of the road geometry is optional. Only attached attributes have to be transmitted, in case of the availability of a digital map within the vehicle. But since a local map might be outdated, a practical solution is the transmission of the complete eHorizon. Therefore, we investigate higher order polynomials for road geometry description, in order to reduce the required data size. We evaluate the compactness of the eHorizon path geometry description in Section

6.4. Finally we conclude this chapter with a discussion and short description of the architectural design and mechanisms in Section 6.5. We have released the source code of *Horizon.KOM*, *RemoteHorizon.KOM* and the proposed data format under the Apache 2.0 open source license.

6.1 HORIZON.KOM - THE LOCAL EHORIZON PROVIDER

The model of *Horizon.KOM* is designed for static and dynamic data as input. Data sources can be vehicle sensors, a local database, or external sources. The most important dynamic data source is position information, which is typically Kalman filtered GNSS data.

The general structure of *Horizon.KOM* consists of six major components. The *tree-Structure (I)* is the central element and basic data storage of the current eHorizon. Based on the vehicle velocity, the *eHorizon Size Determination (II)* component calculates the required eHorizon size. The *Most Probable Path (MPP)(III)* component estimates the most likely driving path, i. e., the main path in the eHorizon. The *Dynamic Road Administration (IV)* component manages the up-to-dateness of the *tree-Structure*. The *Map Matching (V)* component determines the road segment, the ego vehicle is currently driving on, i. e., match the current position onto a road segment. The *Traffic Data Manager (VI)* integrates dynamic external data into the *tree-Structure*, e. g., traffic data. In the following, we give a detailed description of these six components.

(I) Tree-Structure: This is the central dynamic data storage of the eHorizon. Road segments in driving direction are stored into a *tree-structure*, to enable efficient storing with a low processing effort. This *tree-structure* is easy to adapt to changes in the eHorizon, due to vehicle movement. Road segments are represented as nodes, that are connected with edges. Each edge corresponds to an egress of an intersection. The tree's root node is always set to the road segment, the ego vehicle is currently driving on. All road segments, connected to the respective road segment of the root node, form a second-tier to the *tree-structure*. In this multi-tier representation, a node is always accessible by a node of a tier above, connected with an edge, that represents the intersection. We update the *tree-structure* each time the ego vehicle moves onto the next road segment, which then becomes the new root node. Thus, passed nodes are automatically dropped, since no pointer towards the element exists anymore. In case of our Java implementation, the respective object deletion is then performed

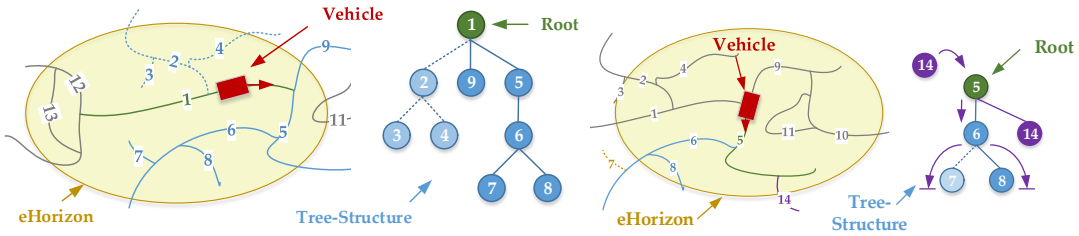

Figure 23: Illustration of the eHorizon and the respective mapping into the tree-structure. Road segments are represented as nodes. The road segment, the ego vehicle is currently driving on, is always the root node. Passed nodes are dropped and new nodes are added according to the vehicle movement. [155].

automatically by the garbage collector. The size of the *tree-structure* and thus, of our eHorizon, is limited by a predefined maximum number of tiers and a maximum length along the MPP. If the ego vehicle passes an intersection, all passed nodes are marked as 'not accessible', but remain in the *tree-structure* until the next update. An example of this update process is depicted in Figure 23. In the illustration on the left side, the ego vehicle has passed *node* 2, thus, *nodes* 2, 3 and 4 are marked as 'not accessible', indicated by a dashed line, but remain in the *tree-structure*. In the illustration on the right side of Figure 23, the ego vehicle has moved onto *node* 5, that becomes the new root node. Within the respective update process, the previous root node and all connected nodes are dropped and new nodes are added, according to the maximum eHorizon size. In the depicted example, *nodes* 1, 2, 3, 4 and 9 are dropped and *node* 14 is added.

In *Horizon.KOM*, road segments are represented within a node as 1D line with a defined length. All intermediate shape points are removed and thus, the amount of data is reduced. A more detailed road geometry can be added as optional additional attributes, related to the line, by a given offset. The basic approach is to add the road shape, by adding curvature values as attributes. This simplification is according to the ADASIS specification, which would enable an easy translation into the ADASIS protocol. All positions on a road segment are stated as relative positions by a distance offset from the starting point of the respective road segment. The minimum description of a road segment is its length, the road type and the curvature values. Additional attributes, like information of traffic signs, are stored in the same way.

(II) eHorizon Size Determination: The length of the eHorizon is determined along the MPP. In case the MPP was not correct, passed nodes are dropped and new nodes have to be updated. Moreover, data storage and processing capabilities are always limited to a certain amount. Thus, the size of the eHorizon should be limited to prevent unnecessary updates of the tree structure. Another aspect is the different information need of applications, according to the vehicle context, in particular for the vehicle velocity. In case of driving with high velocity, e. g., on a motorway, the eHorizon length has to be much larger than in case of driving in urban areas. Therefore, we propose two mechanisms to determine the eHorizon length. The first approach makes use of information about the currently used road type, available from the digital map database, and adapts the eHorizon length to a predefined value. The second approach determines the eHorizon length as function of the ego vehicle's velocity. The aim of this approach is to ensure eHorizon availability at least for a time span δt. But even on the same type of road, the required eHorizon length might strongly differ, e. g., because of speed limits. Within an urban environment an eHorizon length of a few hundred meters is sufficient, whereas driving with high velocity requires an eHorizon length of several kilometers. The ADASIS specification defines a maximum eHorizon length of 8190 m (c.f. Chapter 3.4), that we also set as the maximum eHorizon length. We derived Equation 15 to adapt the eHorizon length L_{MPP} to the vehicle velocity, that is indeed the length of the MPP.

$$Max(L_{MPP}(v), 1km) \quad with \qquad L_{MPP}(v) = 120s \cdot v \tag{15}$$

Thus, the length of the eHorizon corresponds to a remaining driving time of two minutes. This results in about 8.3 km at a maximum considered velocity of 250 km/h, i. e., a similar value as defined in the ADASIS specification. However, we have set the

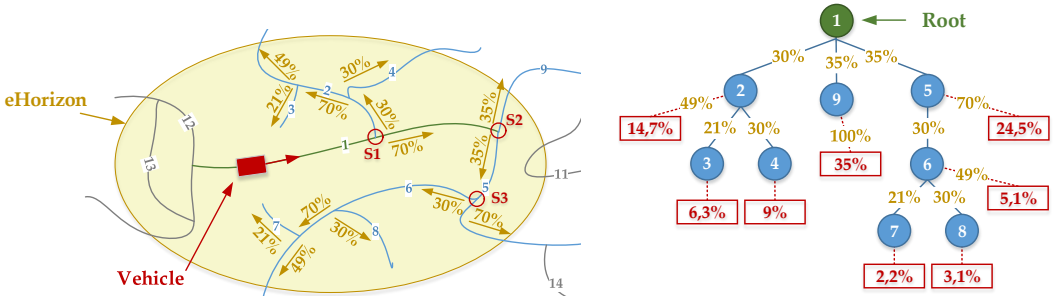

Figure 24: Illustration of turn probabilities with the respective representation in the *tree structure* [155].

minimum length of L_{MPP} to 1km, to ensure also a long-range view at a lower velocity. As an example, this results in 1km at a velocity of 50 km/h or stop and about 3.3 km at a velocity of 100 km/h. Nodes that are already within the *tree structure* are retained in case of a reduced velocity, to prevent high fluctuations. Another aspect, that strongly affects the eHorizon size is the *depth*, i. e., the number of tier-levels, that we assume to be predefined to a fixed value. In summary, the eHorizon length is adapted according to the vehicle velocity, along the MPP with a fixed *depth* of side paths.

(III) Most Probable Path: The MPP is the most likely path, the ego vehicle will take. It is the main path in the *tree structure* and determines the size of the eHorizon. All child nodes of any node in the *tree structure* are possible turn paths in driving direction. The turn probability at each intersection point determines the route of the MPP, that goes along the highest turn probabilities, i. e., traversing the tree along the highest probabilities. In our *Horizon.KOM* eHorizon prototype we assume these turn probabilities to be predefined within the map database, externally given by a navigation system or based on historical driving information (c.f. Chapter 3.4). A mechanism to determine the MPP is not part of *Horizon.KOM*, but the respective properties are considered to be stored within the *tree structure*. An example is illustrated in Figure 24, based on the previously introduced example. A probability to select a specific path is further divided at each following intersection. The vehicle in Figure 24 is driving on *node 1*. It will continue on *node 1* at intersection S1, with a probability of 70%. At S2, the turn probability is equal distributed and thus, the probability of 70% to reach S2 is distributed to a respective turn probability of 35%, to turn on *node 9* or *node 5*. The following turn probabilities are derived accordingly.

(IV) Dynamic Road Administration: The dynamic management of the nodes, within the *tree structure*, is performed by the *dynamic road administration*. The size of the eHorizon is determined by a function of the vehicle velocity. At the minimum, the *tree structure* always consists of a root node, i. e., the road segment, the ego vehicle is currently driving on. Each time the ego vehicle turns onto a new road segment, the tree structure is updated. The *dynamic road administration* selects new nodes from the map database and adds them to the *tree structure*, according to the determined eHorizon size. The component traverses the *tree structure* along the MPP and adds nodes, according to the calculated length. At intersections nodes are added according to the predefined *depth*. Nodes of already passed road segments are dropped.

Figure 25: Screenshot of the Android prototype of *Horizon.KOM*.

Moreover, additional data from other sources is attached as attributes to the respective nodes.

(V) Map Matching: Typically, localization information derived from GNSS suffers from a certain inaccuracy. In case of the availability of local sensor data, including steering angle and velocity, a kalman filter can be used to reduce the potential positioning error. Once a certain position accuracy is achieved, the matching of the ego vehicle position onto the respective road segment brings a further advantage in accuracy. After the map matching, the current position can be described using the *ID* of the current road segment and an offset, i. e., the distance to the start point of the segment. The position history is used to reduce the chance of a wrong matching. Our *map matching* component provides a simplified implementation of the approach, proposed by Quddus et al. (c.f. Chapter 2.7) [47]. The *map matching* component triggers the update process for the *tree structure*, in case of a turn onto a new road segment, i. e., a new root node.

(VI) Traffic Data Manager: The *traffic data manager* enables the dynamic attachment of external information. Such information is attached as attributes towards the respective nodes within the *tree structure*. It is also capable to link a respective information towards several nodes, e. g., a traffic jam, that affects several road segments. Since the prototype implementation of *Horizon.KOM* has not integrated any external data sources, this component serves as an interface for future extension.

The prototype implementation of *Horizon.KOM* is realized as Android application. With regard to a digital map data source, we make use of the Open Street Map (OSM) project. The desired map extract is stored in a local *SQLite* database on the Android device. Since we had no direct interface to the vehicle available, the turn signal input is realized as control buttons. A screenshot of the Android prototype of *Horizon.KOM* is depicted in Figure 25. Road segments, that are within the current eHorizon are shaded in light blue. We have introduced our prototype in [155] and released the source code under the Apache 2.0 open source license. The source files are online

available: `http://www.kom.tu-darmstadt.de/research-results/software-downloads/software/horizonkom/`.

6.2 REMOTEHORIZON.KOM - THE CLOUD-BASED EHORIZON PROVIDER

The local eHorizon provider still lacks of up-to-date data and would require frequent database updates. To overcome this issue, we presented the cloud-based eHorizon provider *RemoteHorizon.KOM* [147]. We have transformed the eHorizon provider concept into a distributed client-server architecture, with asynchronous data transmission. The concept is realized in a publish-subscribe fashion and an overview is depicted in Figure 26. To store and manage the eHorizon, the *tree structure*, as introduced in Section 6.1, is used. The general working principle of the eHorizon also makes use of the MPP. Sensor information, that is required from the mobile clients, i. e., the vehicles, is its *current geographical location*, *velocity* and the *heading*, i. e., the relative angle of driving direction towards true north. For the communication between the backend and the mobile clients, we assume a packet-switched data connection with a relatively constant connectivity, e. g., a *LTE* cellular link. We have named our overall communication system *CarConnect*. The mobile client implements the *CarConnect*, that consists of the components *Communication Core* and *eHorizon Core*. These components enable the remote request of an eHorizon from the backend. The backend implements the respective complement, the *CarConnect Server*, that consists of the components *eHorizon Core*, *Communication Core* and *Data Transmitter*. These components enable the provisioning of a remote eHorizon towards the mobile clients. The backend has access to a digital map database, that is centrally updated, e. g., by a mechanism as proposed in Chapter 5. Optionally, additionel information sources can be integrated.

The general information flow works as follows: If an eHorizon is requested from the *CarConnect Client* on the vehicle side, the current vehicle position, velocity and heading is passed to the *eHorizon Core*. In a next step, this component sends a request to the *Communication Core*, that is connected with an associated *CarConnect Server* backend. On the backend side, the eHorizon request triggers the instantiation of a remote eHorizon. The generated eHorizon is then passed to the data transmitter,

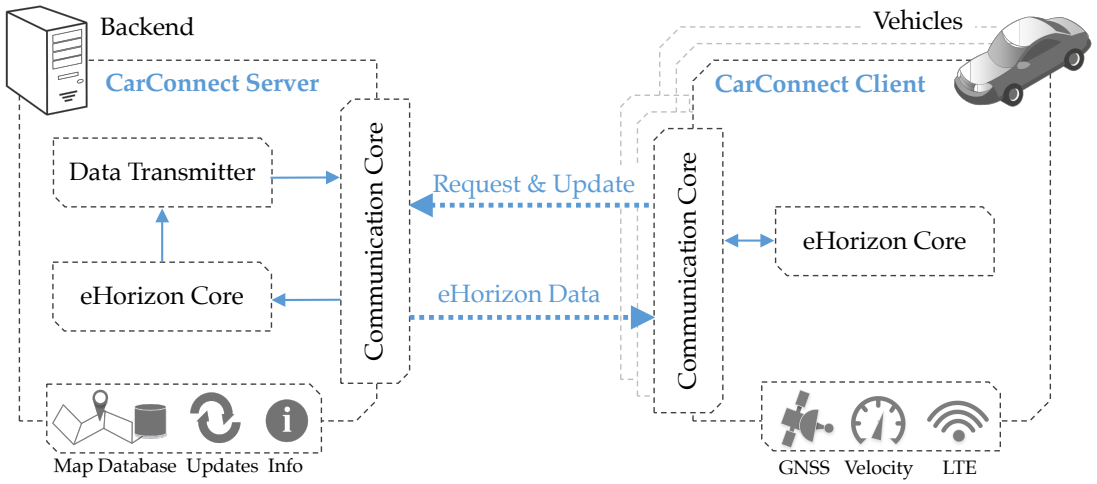

Figure 26: System architecture of the *RemoteHorizon.KOM*.

that disassembles the eHorizon into smaller data packets, depending on the total data size. The data packets are transmitted towards the vehicle via the *Communication Core*. The disassembling into small data packets enables a robust communication, even in case of a low data rate connection with interruptions. As soon as the first data packets have been arrived on the vehicle side, the *eHorizon Core* starts to reassemble the eHorizon. With each received additional data packet, the eHorizon is further completed. Moreover, the communication between the vehicles and the backend is realized via a publish-subscribe middleware. In case of an interruption, the client simply connects again, as soon as the mobile network is available again. Afterwards, the transmission of the data packets continues. While the vehicle is moving, it provides position updates towards the backend. Based on these position updates, the *eHorizon Core* on the backend side is synchronized and updated. These updates are then transmitted to the vehicle side as described before. The update interval depends on the remaining eHorizon length on the vehicle side. In the following, we give a detailed description of each component.

Communication Core: The *Communication Core* is one of the major components. It unifies the communication for the data exchange between the vehicles and the backend. Therefore, it makes use of an Message Queuing Telemetry Transport (MQTT) protocol based publish-subscribe middleware, also named MQTT message broker. For our prototype implementation we used the lightweight and open source MQTT message broker *MOSQUITO*[3]. However, in case of a large deployment, highly scaleable message brokers, like the *IBM WebSphere MQ*, are available. To realize a direct addressing of each single vehicle, the Vehicle Identification Number (VIN) is used, as an individual publish *topic*. The VIN is part of the eHorizon request, which enables the backend to respond to the correct vehicle, without maintaining a direct connection.

The *Communication Core* itself is used on the backend side, as well as in the mobile clients, i. e., vehicles. Sub components are the *message buffer* and the *notification system*. A received message is directly added to the buffer, which enables to cope with input data bursts. Moreover, a copy of the *message buffer* is maintained in the local file system, to enable a fast recovery after system breakdown. The *notification system* component is an internal publish-subscribe system, within the *Communication Core*, to distribute incoming messages towards the corresponding applications, e. g., an instance of the *eHorizon Core*. The *message buffer* is also used for outgoing messages, that are at first buffered and then transmitted to the respective message broker.
Message Structure: For the description and serialization of respective messages, we have defined a *data collection and eHorizon serialization structure*. For the implementation we have used *Google Protocol Buffers (protobuf)*[4]. According to Sumaray and Makki, *protobuf* is the most efficient data serialization format for mobile platforms [156]. Moreover, it is backwards compatible, i. e., a data structure can be extended without the need of updating all clients. A major component of our proposed data structure is the mapping of an eHorizon. But it is also capable for sensor or event data requests and the respective response. Moreover, it is also used for the eHorizon request. The complete communication between the *CarConnect Client* and the *Car-*

3 https://mosquitto.org
4 https://developers.google.com/protocol-buffers/

Connect Server components is exclusively done with this data structure. We have introduced our proposed data structure in [130] and give a detailed description in Chapter A.2.1.
Message Buffer: The main intention of the *message buffer* is to prevent a loss of data in case of bursty data traffic, e. g., in case of many parallel requests at the backend. The *message buffer* stores all incoming messages in a queue, in the working principle of *first-in-first-out*. To prevent a data loss at the vehicle side, a copy of the *message buffer* is stored in the local file system. If the vehicle is powered off, collected data that has not been transmitted can then be uploaded after the next start. The next step, after shifting the incoming message into the *message buffer*, is to trigger the *notification system*. For outgoing messages, the *message buffer* provides an *as-soon-as-possible* approach and an additional delayed transmission mode. In case of the latter one, a transmission time has to be specified. This can be useful to realize a delayed upload of vehicular sensed data, e. g., via a local *Wi-Fi* at home.
Notification System: The *notification system* processes all messages within the *message buffer* and distributes them to the respective applications. Applications can subscribe to a specific type of message at the *notification system* and then get triggered, in case of a respective new message. Supported message content types are: *EHORIZON*, *REQUESTINFO*, *INFO*, *SENSORREQUEST*, *EVENTREQUEST*, *EVENTTAG* and *SENSORTAG*, which corresponds to our proposed data transmission structure. Each time a message is forwarded to one or more application instances, it is cleared out of the buffer.

eHorizon Core: The *eHorizon Core* component implements all required functionalities to provide an eHorizon. The component is used by both, the *CarConnect Client* and the *CarConnect Server*. For each requesting vehicle, a separate eHorizon is instantiated in a separate thread, that directly communicates with the *Communication Core*. On the client side only one eHorizon is instantiated. In this case, the *eHorizon Core* also generates update messages, which include the vehicle velocity, heading and position. In case an eHorizon is already available at the vehicle side, the position is already map matched, i. e., described by the road segment *ID* and an offset.

A *map matching* component determines the road segment, the vehicle is currently driving on. In a first step, all road segments close to the vehicle are requested from the map database. Then the *map matching* component compares the vehicle heading with the orientation of each road segment and calculates the euclidean distance. The *eHorizon Core* also contains an eHorizon size calculator. It determines the eHorizon size, based on the current vehicle velocity, as explained in Section 6.1 and given in Equation 15.

The Most Probable Path (MPP) is calculated based on a heuristic, that is described in detail in Section 6.3. It is basically determined by the road class and the relative angle of egressing road segments, according to Equation 18.

Similar as described in Section 6.1, a *path manager* component maintains and builds up the eHorizon *tree structure*. It traverses the eHorizon along the MPP to determine if new nodes have to be added or already passed nodes have to be dropped. In order to transfer the generated eHorizon data to the client, a message worker generates messages out of the eHorizon and transmits these messages to the sending queue of the *data transmitter*. Only new information of an eHorizon is transmitted.

If a client sends an eHorizon request, an eHorizon is instantiated in the *eHorizon Core* and transmitted to the vehicle. The respective thread is maintained until a timeout. In between, the vehicle can send position update messages, which triggers to update the respective eHorizon instance. New nodes within the *tree structure* of the *eHorizon* are transmitted to the vehicle via the *data transmitter*. On the client side, the *eHorizon* works similar. It forwards eHorizon messages with content of the *tree structure*, that has been generated at the server side. New nodes are added towards the local eHorizon *tree structure*. A map matching is then performed locally on the *tree structure*. If the local map matching is not possible, then the entire *tree structure* is dropped. In any case, the latest vehicle velocity, heading and position is sent as update to the backend to request an *eHorizon* update.

Data Transmitter: The *data transmitter* component adds influence to the transmission behavior of transmitting an eHorizon towards a vehicle. It is located in the structure between the *eHorizon Core* and the *Communication Core* on the server side. A generated eHorizon is passed to the *data transmitter*, that is capable to disassemble the eHorizon into several messages. For the transmission, it provides a default mode, that directly transmits the eHorizon as fast as possible, using the *Communication Core*. An optional mode provides the transmission of an eHorizon as background service. The aim is to avoid transmission bursts and thus, sending in a constant data stream, that is not exceeding a maximum throughput. An eHorizon is disassembled into small data packets, e. g., of about 1 KB each. Given a predefined packed delay, e. g., one second per packet, results in a low data rate of about 8 kbit/s. The data packet size and the transmission delay can be freely configured. If an eHorizon update is triggered, because of a position update, the output queue of the *data transmitter* is checked for already passed nodes. These nodes are dropped to avoid redundant transmission.

Adaptive Transmission: The general data packet transmission order of the *data transmitter* relates to the turn probabilities of the MPP. An extension to the reduced transmission of the *data transmitter* is the adaptive transmission, that adds transmission priorities to the node attributes. The *tree structure* of the eHorizon is now disassembled in a way, that each road segment results in a separate message. Moreover, also each attribute information type, e. g., traffic signs, also generates a separate message. Thus, each road segment with n different attribute types, results in $n + 1$ messages. In the following, we consider an example of three attribute types available, with initial type transmission priorities 0.7, 0.5 and 0.2. The transmission priority of the road segment is 1.0. It is the most important information, because the attached attributes

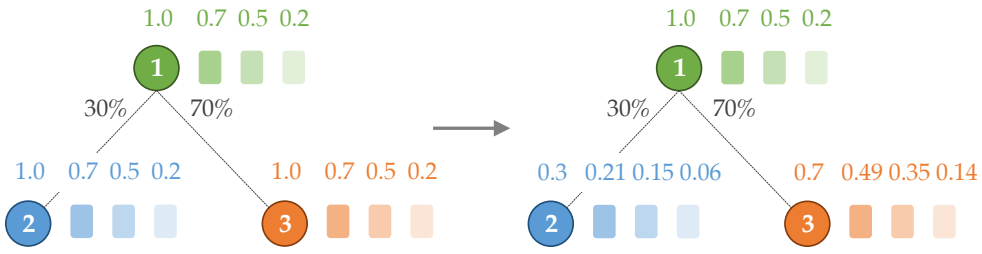

Figure 27: Example of the transmission priority calculation.

Table 10: Comparison of the default transmission and the adaptive transmission, with respect to the total amount of transmitted data.

Approach	**Transmitted data**
Default transmission	18454.7 kbit
Adaptive transmission	3101.5 kbit

relate their positions relatively to the road segment. Such a decreasing transmission priority could as example reflect the importance of *traffic signs*, *parking space info* and *POI* information. In this case, the decreasing importance is obvious. The transmission priorities of all messages are multiplied with the turn probabilities, according to the MPP determination. An example is depicted in Figure 27. The left side of Figure 27 depicts three nodes, each with three attaches attribute types. This results in a total of twelve messages. The initial transmission priority of each message is always denoted above. Node 1 is the root node. The turn probability is 30% to select node 2 and 70% to select node 3. The right side of Figure 27 depicts the resulting transmission priorities. In the output buffer, all messages are reordered according to the given transmission priority. A higher priority number reflects an earlier transmission. This concept, in combination with the low data rate transmission mode of the *data transmitter*, results in dropped information of not used road segments. Each time the vehicle turns onto a new road segment, i. e., the root node changes, a position update is provided to the backend. As a result, all messages that are related to already passed road segments are dropped from the transmitter queue. Hence, the desired transmission bandwidth controls the ratio of dropped messages. This is especially beneficial, in case of an unknown route and no historic route information about the driver. In such a case, the MPP has to be determined by a heuristic. Since this mechanism can not always predict turns correctly, this approach enables to reduce the total amount of transmitted data. However, the overall performance strongly depends on the desired transmission rate and the actual map database.

To get an impression of the described mechanism behavior, we have conducted a test drive with example data. Each road segment has three attribute types attached, with the same transmission probabilities as mentioned before. The data size of each attribute message has been set to a fixed value of 300 Bytes. The data size of the node message, i. e., the road geometry, depends on the data from our OSM based digital map database. The test drive started in Rüsselsheim, Germany and ended in the inner city of Darmstadt, Germany. The total length of the route was about 30 km, including mostly motorway and urban roads. The desired transmission rate has been configured to 5 kbit/s. We have measured the total amount of transmitted data. The result is given in Table 10, in comparison to the default transmission mode. We can see a reduction of about 80% in the total amount of data traffic. However, this is mainly because the desired transmission rate has been set to a very low value, which strengthens the effect to drop messages before sending. We have observed some missing information of departures on the motorway in this example test drive. Due to the high velocity, the transmission rate was not sufficient to deliver all messages related to possible turns, before passing by. A more detailed evaluation would

require a large set of realistic data, i. e., a high detailed digital map. However, this test still confirms the feasibility of our approach.

6.3 GENERIC MPP DETERMINATION

The MPP has significant influence to the eHorizon quality [6]. In case of using a navigation system, the MPP can be directly derived from the calculated route. Else, historic mobility data can be used to estimate the MPP [6, 122]. However, in some cases both is not available. In such a case, the MPP has to be determined by a heuristic. We assume the only available input for the algorithm is the initial vehicle position, including the heading and a map database. The map database contains information of the road class, i. e., the type, and the geometry of each road segment. Thus, we derive a heuristic, based on the relation of the road class of an approaching road segment to the egress road segments and the relative angles to each other. The assumption is to prioritize faster road types, i. e., prioritize a motorway to a normal road, and roads with a smaller turning angle. For each road segment Seg_i, we calculate a weighting Θ_{RC_i} for the road class and a weighting Θ_{γ_i} for the turning angle. Based on both, we calculate a probability $p(Seg_i)$, to turn into the respective road segment Seg_i. We use the road classes as given in the OSM map material, whereas a lower number indicates a faster road type, i. e., a motorway has a lower number than an urban road. The weight Θ_{RC_i}, of a branching road segment Seg_i, is the fraction of the road class of Seg_i and the sum of the road classes of all N branching road segments of the respective intersection. The calculation of Θ_{RC_i} is given in Equation 16. Since Θ_{RC_i} should increase for faster road types, we subtract the fraction from one.

$$\Theta_{RC_i} = 1 - \frac{RC_i}{\sum_{n=1}^{N} RC_n} \quad ; \quad \Theta_{RC_i} \in [0..1] \tag{16}$$

The angle γ_i of an egress road segment Seg_i is the relative angle between Seg_i and the approaching road segment. The relative angle $\gamma_i \in (0°, 180°)$ of two road segments can be easily calculated by the magnitude of the difference of each road segments angle, towards true north. The weight Θ_{γ_i}, of a branching road segment Seg_i, is the fraction of the relative angle γ_i, to the sum of the angles of all N branching road segments of the respective intersection. The calculation of Θ_{γ_i} is given in Equation 17. Since Θ_{γ_i} should increase for smaller relative angles, we subtract the fraction from one.

$$\Theta_{\gamma_i} = 1 - \frac{\gamma_i}{\sum_{n=1}^{N} \gamma_n} \quad ; \quad \Theta_{\gamma_i} \in [0..1] \tag{17}$$

The respective turn probability of a road segment Seg_i is a function of the calculated parameters Θ_{RC_i} and Θ_{γ_i}. In addition, a factor α weights the importance of the road class and a factor β weights the importance of the relative angle. The relation of α and β determines the relation in the weighting of Θ_{RC_i} and Θ_{γ_i}. The calculation of the probability p_{Seg_i}, to turn into road segment Seg_i, is given in Equation 18.

$$p_{Seg_i} = \frac{\alpha \cdot \Theta_{RC_i} + \beta \cdot \Theta_{\gamma_i}}{\sum_{n=1}^{N} (\alpha \cdot \Theta_{RC_n} + \beta \cdot \Theta_{\gamma_n})} \tag{18}$$

$$\alpha \in [0..1] \quad ; \quad \beta = 1 - \alpha \quad ; \quad p_{Seg_i} \in [0..1]$$

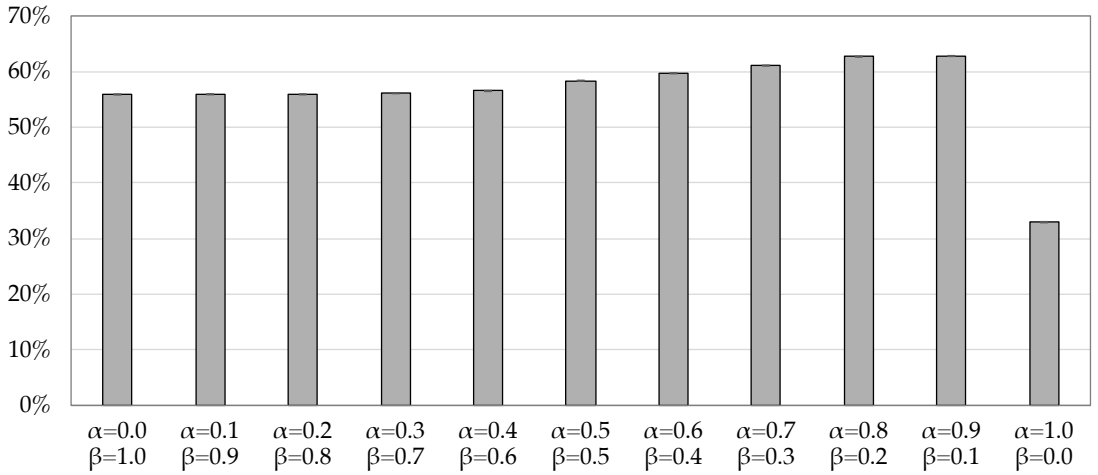

Figure 28: Correctness of our MPP heuristic in the scenario of *TAPAS Cologne* for different values of α and β.

Parameter determination: Our heuristic to determine the MPP is solely based on digital map parameters. The assumption is, that the road angle and the road class of egressing road segments at an intersection condition the turn behavior to a certain extent. To determine the relation of the weighting of the road angle and the road class, we analyzed the behavior of different parameter values of α in a simulation. Therefore, we used the open traffic simulation suite SUMO[5] for the simulation of the road network and vehicular traffic, introduced in Chapter 5.3.1.1 [150]. The tool allows to simulate the traffic flow within a map-based scenario. We have used the freely available simulation scenario of Cologne [152]. The scenario is based on OSM map material and the *Travel and Activity Patterns Simulation (TAPAS)* model [152]. Thus, the scenario provides realistic traffic patterns. The freely available part of this scenario consists of two hours traffic, that is constantly increasing to a maximum of about 13300 vehicles running in parallel. In total, the scenario contains about 75000 individual routes. We have used this set of vehicle routes to determine the correctness of our MPP heuristic, for different values of α and β. We define this correctness as the ratio of correct turn decisions, to the total number of turns. The respective results are depicted in Figure 28. Confidence intervals were computed using Student's t-distribution [154]. In addition to Figure 28, the results are given in Section A.3.2 in Table 26. Most values of α and β show results above 50%. These are good results, since the baseline is below 50%, because of most intersections have three egressing roads. This shows, at least in this scenario, our proposed algorithm is able to determine the MPP with an accuracy larger than 50%. The best result, of about 62.8%, is achieved in a combination of $\alpha = 0.9$, i. e., $\beta = 0.1$. This shows, that the road class has a much stronger effect, but completely discarding the influence of the road angle, i. e., $\alpha = 1.0$, i. e., $\beta = 0.0$, shows a significant drop in accuracy. However, this might be due to the simulation scenario, that reflects a morning rush hour. In other scenarios the influence of the road angle might be stronger.

We have used the parameter combination of $\alpha = 0.9$, i. e., $\beta = 0.1$, to evaluate the behavior of our MPP heuristic, in the use of our remote eHorizon. First, we have executed the scenario and extracted all route courses for further analytics. We have

5 http://sumo.dlr.de/

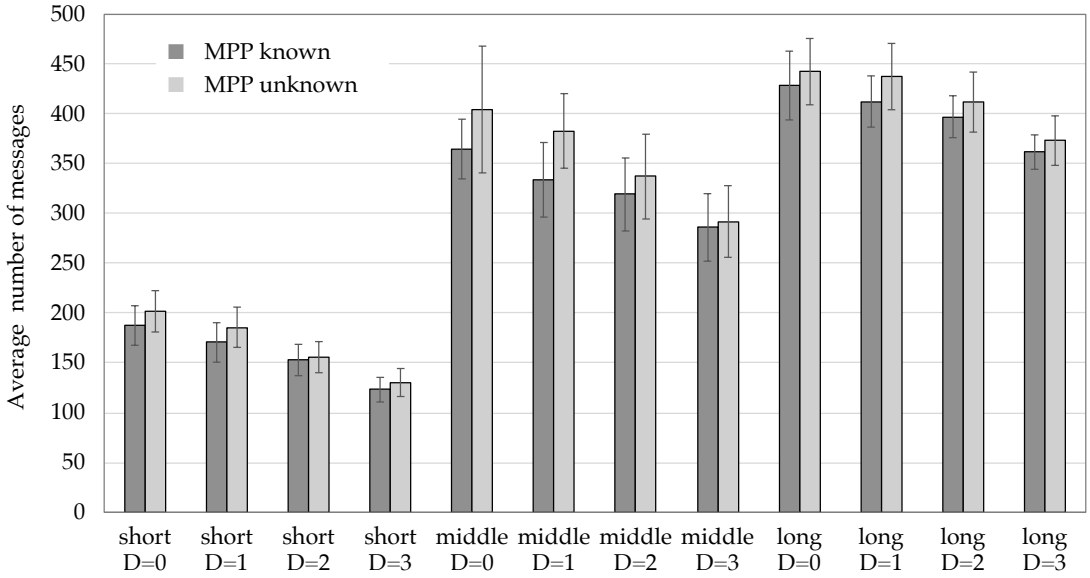

Figure 29: Average amount of messages, in comparison of a completely known MPP and the usage of our MPP heuristic with $\alpha = 0.9$, i. e., $\beta = 0.1$. Each, for an eHorizon depth of $0, 1, 2$, and 3 and the three selected groups of traces.

ordered the 75000 individual routes of the scenario, according to the total trip length. Next, we have divided the total set of traces into three groups, according to the trip length. We have selected 10 km and 40 km as partition limits and used the 200 longest trips of each of these three groups, to evaluate the MPP heuristic in our remote eHorizon. Since the 100% correct MPP is known from the route courses, we use this known MPP as baseline. We compare our MPP heuristic against the completely known MPP in the use of our remote eHorizon. Each for an eHorizon depth of $0, 1, 2$, and 3 and the three selected groups of traces. As a metric, we compare the necessary average amount of messages. The respective results are given in Figure 29. Confidence intervals were computed using Student's t-distribution [154]. In addition to Figure 29, the results are given in Section A.3.2 in Table 27. The eHorizon length is in all cases determined as described in Equation 15 in Section 6.1. As expected, the amount of messages increases in all cases, with an increasing path length. Moreover, an increased eHorizon depth reduces the total amount of messages, because the resulting eHorizon contains more road segments, i. e., the probability is high, to already include necessary road segments. This is also true in case of a known MPP, because always only a subset of the complete eHorizon, according to the calculated length, is transmitted. Thus, at the end of a transmitted eHorizon segment, the side paths correspond to the following MPP. Overall we can see, that in case of the MPP heuristic, the amount of necessary transmissions approximates the respective value, in case of a known MPP for an increased eHorizon depth. But simultaneously, the amount of transmitted data is tenfold increasing compared to a known MPP.

In general, we can assume the MPP to be known from historic data or from the navigation system [6, 122]. In case of the MPP is not known, the use of our presented heuristic still enables the robust use of the eHorizon, but with an increased amount of data traffic.

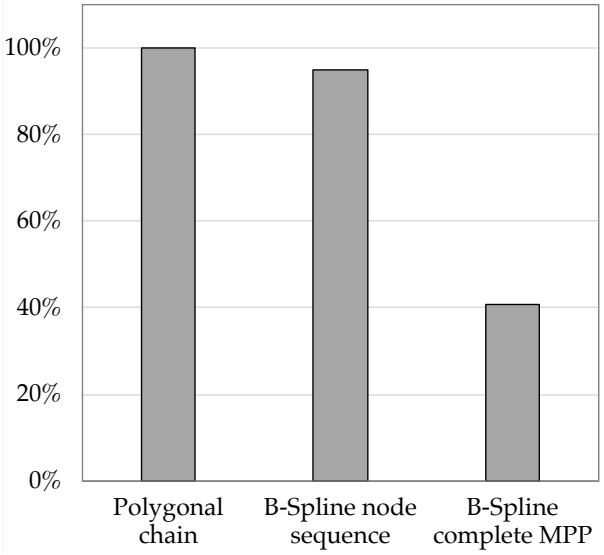

Figure 30: Comparison of the possible reduction in data size of the road geometry in the eHorizon, by the use of B-Splines.

6.4 PATH DESCRIPTION AND COMPACTNESS

In order to decrease the overall data size of the eHorizon, the description of the path geometry can be optimized, since this is the most significant contributor to the data size. As already introduced in the ADASIS specification, the road geometry can not only be described as a linear interpolation of points, but also by a higher order interpolation profile [24]. Thus, we added to our data structure the option, to model the road geometry within eHorizon nodes as B-Splines. The ADASIS specification uses a fifth order B-Spline, i. e., a B-Spline degree of four (c.f. Chapter 2.6.2), to model the road shape. This would result in a minimum of five control points, to model the respective B-Spline. However, the OSM map material of the used *TAPAS Cologne* scenario is not detailed and most road segments only consist of a few shape points. The complete scenario consists of about 45000 road segments, but only about 6.5% of the segments are composed of eight or more shape points. Thus, we selected a B-Spline degree of only three, to model the road geometry. This requires a minimum of four control points. We extended our remote eHorizon to model all segments, composed of eight or more shape points as B-Spline. The number of control points was set to 50% of the number of shape points, to reduce the amount of data for road geometry description. The available map material has not enough points to significantly reduce the amount of data, but nevertheless we like to highlight the optimization potential. Thus, we compare the relative possible reduction of the total eHorizon data size, of the previously mentioned longest 200 trips of the *TAPAS Cologne* scenario. The result is depicted in Figure 30. The baseline is the total eHorizon data size, in case of not using B-Splines. The eHorizon depth was set to zero, i. e., the eHorizon only consists of the MPP. In the middle of Figure 30, we can see a reduction in the total data size of about 5.1%. This reduction is for substituting the path description as B-Spline of road segments with eight or more shape points. On the right of Figure 30, we can see a reduction in the total data size of about 59%. This reduction is for substituting the whole MPP with a single B-Spline. In this extreme example, all segments of the

MPP have been concatenated and substituted by a B-Spline. The amount of control points is 50% of the amount of the original shape points. The resulting reduction is more than 50%, since typically the end point of a road segment is the same as the start point of the succeeding road segment. Thus, the concatenation of N segments results in $N - 1$ less shape points. Due to the potentially high reduction in the data size, we propose the use of B-Splines for road geometry description, in case of high detailed map material.

6.5 DISCUSSION AND SOURCES

The use of an eHorizon is an appropriate mechanism to provide a long-range perception to ADASs [5]. An eHorizon is suitable to represent static and semi dynamic location based information. The data source is typically a local map database, that has to be frequently updated, in order to provide a correct eHorizon [123]. To overcome this and make frequent map updates unnecessary, we propose a cloud-based approach. A logically central eHorizon provider is able to always provide up-to-date information. Within this chapter, we have introduced our concept of a remote eHorizon, that can serve as cloud-based eHorizon provider. Our concept also includes a heuristic MPP determination, which allows to provide an eHorizon without any knowledge about the requesting vehicle. However, knowledge about the MPP from historic data or from the navigation system allows to decrease the necessary amount of data transmitted. Our proposed remote eHorizon is also already enabled to provide high-resolution maps by the use of B-Splines.

Moreover, we have provided the source code of our prototype under the Apache 2.0 open source license online to the community: `http://www.kom.tu-darmstadt.de/research-results/software-downloads/software/horizonkom/`. The provided components consist of a stand alone local eHorizon provider, as well as the remote eHorizon. In addition, we provide all sources of our proposed data structure, that enables an efficient eHorizon transmission, but can also serve for remote data acquisition in the vehicular domain. The aim is to provide a framework for an easy development of future vehicular applications. The modular implementation makes use of multithreading and the communication is based on the publish-subscribe paradigm. This already ensures a high scaleability, even in the prototype status. Remote ADAS and infotainment applications can be easily developed on top of our framework. The asynchronous communication concept enables a robust communication, even in case of a connection with interruptions. In addition, our framework enables the data transmission as background service, to prevent data bursts. Finally, the developed eHorizon realizes an efficient up-to-date information service, that facilitates novel ADASs and can support autonomous driving features.

7 EXTENDED VEHICULAR PERCEPTION

A key topic for connected ADAS is tracking neighboring vehicles, which is typically realized by a broadcast of ego vehicle information via direct ad-hoc V2V communication. The respective message is defined in the US as BSM in the IEEE 1609 WAVE standard and in Europe as CAM in the ETSI *ITS-G5* standards [36, 18]. Both can carry several optional information, but always contain position, speed, acceleration, heading, steering wheel position and dimensions of the ego vehicle. Moreover, all attributes are provided with certain confidence indicators. Following this, vehicles in communication range are able to track the position and path of the sending vehicle. If the amount of vehicles in proximity increases, the broadcast of position beacon messages, i.e. CAMs, interfere with each other. In this case, respective DCC mechanisms reduce transmit power and sending frequency to reduce interference and channel congestion. This causes a reduced tracking accuracy, caused by a reduced CAM transmission rate and in particular limits the information broadcast to the single hop communication range. In this chapter we present a system to reduce the channel load for object tracking and to extend the vehicular perception range beyond the direct communication range. The according use cases are related to the *awareness zone* as described in Chapter 4.3.

The remainder of this Chapter is structured as follows: In the next Section 7.1 we describe the need of an according system and our objectives, followed by the description and overview of our system in Section 7.2. In the subsequent Section 7.3 we provide our evaluation, beginning with a description of our evaluation setup, followed by the evaluation metrics. Finally we discuss our evaluation results and findings.

7.1 PROBLEM STATEMENT AND OBJECTIVE

Tracking of moving objects in the vehicle surroundings, in particular in driving direction, is an essential foundation for ADAS systems with regard to the *safety zone* and the *awareness zone*, as introduced in Chapter 4. While use cases related to the *safety zone* are covered by local sensor information and mobile ad-hoc communication in single hop range, *awareness zone* related use cases require information forwarding. Such use cases are in particular cooperative driving maneuvers, as described in Sections 4.3 and A.1.2. It is important to have information about moving objects in a distance beyond direct communication range. Long-range driving adaption is important for ADAS, supporting highly automated driving maneuvers or even fully automated driving. This enables coordination between vehicles, to ensure smooth traffic movement, without jumpy trajectories and concertina effect.

As mentioned before, respective message formats, designed for object tracking in vehicular ad-hoc networks, are CAM and BSM [36, 18]. Based on the used congestion control system, these messages are broadcasted with a frequency between 1 Hz and 10 Hz. In general, channel congestion should be avoided. For congestion control

are currently two general approaches are in discussion, a reactive and an adaptive approach. In its specification, ETSI presents the reactive approach, named Decentralized Congestion Control (DCC), which is based on the measured channel load [39]. Moreover, the message header contains information about the ego vehicle's perceived channel load to inform neighbors. The message broadcast frequency is then controlled according to the known channel load of the own measurement and the received information. Channel load is quantized into five channel load classes, that directly control the message output rate as given in Table 11.

Table 11: Mapping of channel busy rate values to defined broadcast state, including respectively allowed transmission rate according to ETSI DCC specification [39]

State	**Channel load**	**Packet rate**	$\mathbf{T_{off}}$
Relaxed	< 30%	10 Hz	100 ms
Active 1	30% to 39%	5 Hz	200 ms
Active 2	40% to 49%	2,5 Hz	400 ms
Active 3	50% to 60%	2 Hz	500 ms
Restrictive	> 60%	1 Hz	1 000 ms

Beyond the reactive DCC approach, an adaptive approach exists, named *A Linear Message Rate Control Algorithm for Vehicular DSRC Systems (LIMERIC)* [80]. It also collects channel load information about the local neighborhood and tries to determine the number of other vehicles in the area. In contrast to binary congestion control, this approach relies on full precision of all available inputs about the wireless channel. The sending rate is calculated linearly and it is shown that the system converges towards a fair and efficient channel utilization [81]. However, even with sophisticated congestion control mechanisms, the channel load is close to its limits in dense traffic situations, due to the high number of messages. Hence, the minimal sending rate for CAMs has been reduced from 2 Hz to 1 Hz in specification [157]. Moreover, most simulations only consider CAM or BSM, but additional messages like Decentralized Environment Notification Message (DENM) will cause additional load on the channel. This leads to collisions and loss of messages and thus, to a higher position tracking error. For vehicular ad-hoc communication, seven 10 Mhz channels in the 5.9 Ghz band are reserved in the US and the EU [10]. Currently only one control channel is planned for emergency and object tracking relevant information. The other channels are reserved for other services, e. g.,traffic light information or media services. However, even if a complete channel will be dedicated for the use of object tracking, the channel capacity will not be sufficient for multi-hop propagation of object tracking relevant information [143, 158, 76]. Hence, existing DCC approaches try to care the symptoms of the channel congestion problem, but do not target the origin. To identify the problem's origin, we focus on the purpose of the system: a distributed knowledge about vehicle locations in the interfering traffic environment. The standardized mechanisms solve this, sending periodic updates, which contain a lot of redundancy. Transmitting redundant information is the origin of the channel congestion problem. Hence, it is expedient to investigate the question: How can we minimize redundancy in transmitted packets?

In contrast to current approaches, that regulate the sending rate by measurements of the channel load, we suggest to reduce existing redundancy. In our approach we adapt the sending rate by detecting redundant information by predicting the estimated vehicle position. With current approaches, the receiver has to use the outdated received position information in case of high channel load. Alternatively, the receiver is able to extrapolate the sender position out of a constant-velocity / constant-heading coasting model, based on the received position information. We extend this approach by the use of two longitudinal kinematic models, namely a *constant-velocity* and a *constant-acceleration* model, and combine them with an adaptive heading model. We switch between these kinematic models and derive the heading by the use of simple map material. As map material we use Open Street Map (OSM) and to indicate a switch of the kinematic model we send a message. Using this information we use the same prediction module on the sender side as on the receiver side. The sender is constantly observing the deviation of the estimated position of the prediction module and the measured position from the ego vehicle's position sensors. A message is sent out, only in case of an error in the estimated position above a predefined threshold ε, or a necessary switch of the kinematic model. In other words, we check the entropy of the messages. Instead of controlling the sending rate by observing the channel load rate, we reduce the overall message sending rate to prevent a busy channel. With our approach we are able to show a reduction of up to 93 % in sent messages, compared to CAM sending according ETSI standard, including DCC. Moreover, we use freed-up bandwidth to forward information. We have defined a message structure to broadcast information about moving objects in the ego vehicle's surroundings. Received CAM messages from vehicles in the surroundings are managed in a local database and respective position information is continuously updated by prediction or new incoming CAM messages. This information is periodically packed into a list and broadcasted. As a result, we are able to more than double the range of vehicular perception, with respect to moving objects. In the following section, we give a detailed description of our system to reduce channel load for object tracking and to extend the vehicular perception.

7.2 SYSTEM CONCEPT AND SPECIFICATION

Our approach basically consists of two steps to achieve an extended perception of moving objects. First, we reduce the ad-hoc channel load by the use of tracking with prediction. In the second step, we use freed-up bandwidth to propagate perception knowledge, i. e., a list of known moving objects in the surroundings. For this, we have defined a dedicated message structure, the so called Cooperative Perception Message (CPM).

In contrast to known DCC mechanisms, that attempt to reduce the channel load if it gets congested, we proactively reduce the channel utilization to a minimum. Our basic concept is depicted in Figure 31. The sender observes an ego position estimator, named *Remote Estimator* in Figure 31, to rate the prediction accuracy that neighbor vehicles are able to achieve, based on the last sent position beacon. The used prediction module on the sender side is the same as on the receiver side. For realization of the position beacons, we send a modified version of the ETSI specified CAM. A description of V2X message types is given in Chapter 2.4.2 and in Sections A.2.2 and

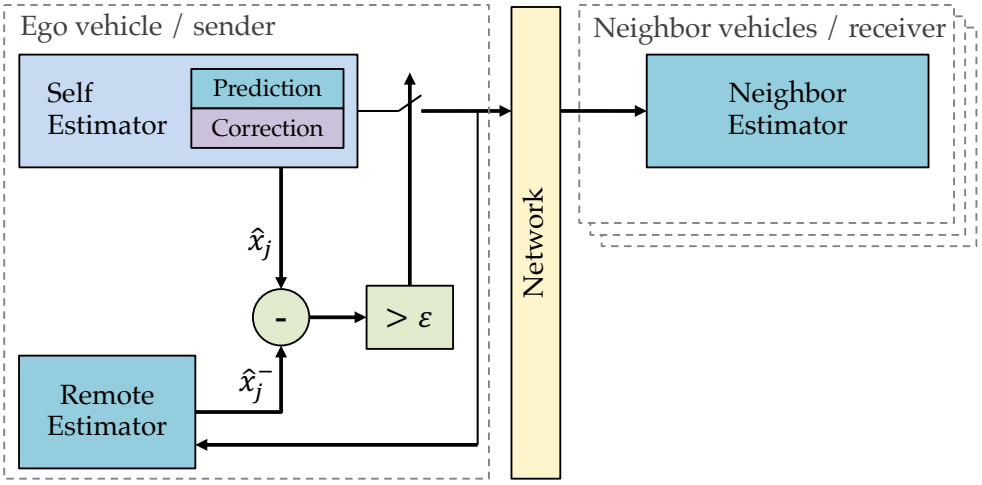

Figure 31: Overview of the system concept to estimate the prediction accuracy of neighbors. A new CAM is sent if the error ε exceeds the threshold, or a timer exceeds (not depicted).

A.2.3 we give details about our modified CAM and our definition of the CPM. For position prediction, we use two kinematic models, namely a *constant-velocity* and a *constant-acceleration* model. In the *constant-velocity* model we define a constant velocity for the respective prediction step and in the *constant-acceleration* model, we define a constant acceleration for the respective prediction step. Which model to use, is decided on the sender side and encoded in the CAM. In addition, we combine the kinematic model with an adaptive heading model. The heading is derived from simple map material, which is OSM data in our case. Vehicles that receive this modified CAM use a position estimator, named Neighbor Estimator in Figure 31, to track the sending vehicle. On the sender side, the magnitude of the difference in the current position $\hat{x}_j$ to the position $\hat{x}_{\bar{j}}$, a neighbor would estimate based on the last sent CAM, is compared against a tolerated error ε, according to Equation 19.

$$|\hat{x}_j - \hat{x}_{\bar{j}}| > \varepsilon \tag{19}$$

The subtraction of the current position and the estimated position, and comparison against the tolerated error ε, are depicted in green in Figure 31. If the error is smaller than the tolerated error ε, no CAM has to be send and the system continues in observing the prediction error. If the calculated error exceeds ε, a new CAM is broadcasted. Basically our system can be divided into several sub components, that are explained in the following.

7.2.1 *Trigger Check If a New CAM Has to Be Sent*

As previously mentioned, we only broadcast a new CAM, if the prediction error on the receiver side exceeds a threshold ε. But in detail, we use four trigger conditions to send out a new CAM. If one condition matches, a new CAM is broadcasted. The first trigger condition is the time that has passed since the last broadcast of a CAM. Here, we have defined a maximum tolerated time $T_{CAM,max}$ that defines the minimal CAM sending rate. The maximum sending rate is defined by $T_{CAM,min} = 100ms$, according to ETSI and WAVE specifications [36, 35]. This also corresponds to the trig-

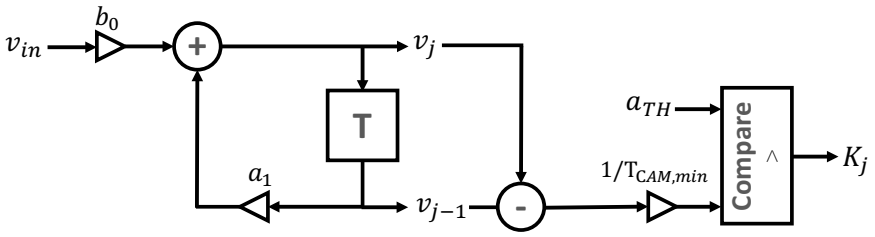

Figure 32: Block diagram of kinematic model selection.

ger interval to check if a new CAM has to be sent. The second trigger condition is a change in the used kinematic model K, i.e., constant velocity or constant acceleration model. Our system observes changes in velocity over time, i.e. a change in the current acceleration. We use a simple first order IIR low pass filter to prevent high frequent changes in the used model, which would cause a high CAM message overhead. The condition to switch the respective model is based on an acceleration threshold a_{TH} and is given in Equation 20 and depicted as block diagram in Figure 32. The used velocity in the current trigger interval v_j is a weighted product of the currently measured velocity and the used velocity in the previous trigger interval v_{j-1}. To prevent high fluctuations, we weight the currently measured value only with $b_0 = 0.1$, but value of the previous trigger interval with $a_1 = 0.9$.

$$\begin{aligned} K \in \{&const_{velocity}; const_{acceleration}\} \\ &b_0 = 0.1; a_1 = 0.9 \\ &v_j = b_0 \cdot v_{in} + a_1 \cdot v_{j-1} \\ &\frac{|v_{j-1} - v_j|}{T_{CAM,min}} > a_{TH} \end{aligned} \tag{20}$$

We calculate the magnitude of the difference of the velocity in the current trigger interval v_j and the velocity in the previous trigger interval v_{j-1}. We consider this velocity difference per trigger interval $T_{predict}$, which results in an acceleration. This resulting acceleration is compared against an acceleration threshold a_{TH}. If it exceeds the threshold a_{TH}, we switch to the *constant-acceleration* model, otherwise we use the *constant-velocity* model.

The third trigger condition is a lane change or a change of the road segment, e. g., if a vehicle turns at an intersection. The lane ID, which also indicates the road segment, is part of our CAM and is used for adaption of the heading in position prediction. This reduces computation effort for map matching and ensures that a receiving vehicle will predict the heading, based on map data of the same lane as the sending vehicle. It is not the most important thing that the lane is determined correctly on the sender side, but that both, sender and receiver predict the heading, based on the same lane data. The fourth and final trigger condition is the previously mentioned prediction error that is observed at the sender side. If the ego vehicle position a neighbor would predict exceeds a threshold ε, a new CAM is broadcasted.

However, the maximum CAM sending frequency is bound to the rate of how often the trigger conditions are checked. Within our work, we have set this to 100ms, which leads to a maximum CAM sending frequency of 10 Hz. This also corresponds to ETSI specification [35]. In our system, we use a modified version of the ETSI specified CAM, that has two additional values in the high frequency container. The first

additional value determines the kinematic model to be used for position prediction. Here, we have used a constant velocity and a constant acceleration model in our system. The second additional value indicates the lane identification number (ID), of the lane the vehicle is currently driving on. This lane ID is derived from the previously mentioned OSM map material, that is assumed to be the same for all vehicles. The respective information is used to derive the curvature value for the calculation of the vehicle heading.

7.2.2 *Handle Incoming CAMs*

Whenever a CAM message from a neighbor vehicle is received, it is saved to a local database. Within the database we keep for each neighbor vehicle the CAM information and an additional representation for position prediction. This allows to keep the CAM information as received, for later comparison. Moreover, the data fields in the original CAM format do not have enough precision for prediction, which would lead to accumulated errors. The prediction loop is triggered every $T_{predict} = 10ms$ (similar to ETSI specifications [35]), for a fine grained adaption of the heading, based on lane changes according to the map material. This is in particular important on curvy road segments. We predict the position and in case the *constant acceleration* model is used, also the vehicle velocity. Moreover, we store an entry to a list to handle knowledge about directly received CAM messages of neighbor vehicles. This allows to distinguish between knowledge about neighboring vehicle information from CAM or CPM messages and is used for distribution of our environment perception model, i. e., Cooperative Perception Message (CPM). If the received CAM from a vehicle is not yet known, we set a flag f_{new} that triggers an earlier broadcast of a new CPM. If we do not receive any CAM update from neighbors for a time longer than $T_{CAM,expired} = 5s$, i. e., a neighbor vehicle is out of communication range, we discard respective information from our database.

7.2.3 *Trigger Check If a New CPM Should Be Sent*

The initial condition to broadcast a new Cooperative Perception Message (CPM), is to check if any neighbor vehicle is already known. We check the flag f_{new} if new vehicles are known in our database, i. e., vehicles that have not been listed in a CPM yet. In this case, a new CPM is broadcasted if the time difference since the last CPM broadcast has a minimum of $T_{CPM,min} = 2s$ and the flag f_{new} is reset. We have set $T_{CPM,min}$ to 2s because this corresponds to our minimum CAM sending rate. In case the flag f_{new} is not set, CPMs are broadcasted with a sending rate of $T_{CPM,max} = 5s$. Since knowledge about neighbor vehicles is similar for closely driving vehicles, the probability of broadly similar CPMs is high. This is the reason to set $T_{CPM,max}$ to a much higher value, compared to the minimum CAM sending interval $T_{CAM,max}$. Our CPM contains a list of objects that represent information about neighbor vehicles. Each object consists of the minimal CAM version, plus the time stamp of CAM generation of the respective sending vehicle and a lane offset. This means that we only use the high frequency container of the CAMs. The additional time stamp is necessary, since the CAM information in the CPM is not as originally received, but calculated via position prediction. As mentioned before, we use position prediction

to estimate the position of the neighbor vehicle, based on the last received CAM. This predicted position is also used within the CPM. The lane offset describes the longitudinal position of a vehicle on the lane. This reduces computational load on the receiver side, but slightly increases the message size by about three additional Bytes. Per vehicle object we have a maximum data size of $CAM_{size,max} = 51$ Bytes.

The maximum payload size of a IEEE 802.11p message with Geo-Networking and Basic Transport Protocol, according to ETSI specification, is 1394 Bytes. Our minimum CPM message without neighbor vehicle information, requires 9 Bytes. Thus, we can send 27 neighbor vehicle information objects with a maximum size of 51 Bytes, within one single CPM. The ASN.1 definition of our CPM is given in Section A.2.2. In case of local knowledge about more than 27 neighbor vehicles, we require a selection criteria. Firstly, we add only neighbor vehicle information into the CPM if we have received a CAM directly of this vehicle. This means, we do not broadcast vehicle information that we only know from received CPMs. The second condition is to use the 27 objects with the largest relative Euclidean distance to the ego vehicle. This is because direct neighbors most probably have information about each other. Selecting the objects with the largest distance corresponds to information relaying of objects outside the direct communication range. However, if payload is still available in our CPM, this correlates to a sparse traffic situation. In this situation, relative velocities between vehicles tend to be higher, which makes a wider information scope more valuable. In addition, the channel load correlates to traffic density and is, therefore, supposed to be low. Hence, to close the situational information gap, we fill up the CPM with vehicle information objects, based on knowledge from received CPMs. Here, we select information about the closest objects, because this reflects information about vehicles closely behind communication range. These are not relayed multiple times and thus the most up to date information. With this approach we are able to extend the perception range by 2 to 3 times, as shown in the following evaluation.

7.2.4 *Handle Incoming CPMs*

When a vehicle receives a CPM, several conditions have to be checked for all CAM items within the CPM. If an object is already known in our database, we check the generation time stamp of the original CAM. If the time stamp in the CPM is more recent, we transfer the object update to our database and update the respective status as received by CPM. Else, the object update information is discarded. If the object is not known yet and CAM generation time is not older than $T_{CAM,expired} = 5s$, we insert it to our database.

7.2.5 *Prediction Handling*

To begin with, the prediction step interval $T_{predict}$ is set to $10ms$, to achieve a fine grained adaption. In the first step of our prediction mechanism, we check if CAM generation time is older than $T_{CAM,expired} = 5s$. In this case, the respective information is discarded from our database. For the remaining objects we execute a prediction step. We start to update the heading, based on the underlaying map material, by calculating the relative angle θ_j to true north of the current road segment.

Thus, we first have to match the correct road segment. In the next step, we calculate the relative distance d_j, driven since the last update, based on the current kinematic model.

$$\begin{aligned} d_j &= v_j \cdot T_{predict} + \frac{1}{2} a_j \cdot T_{predict}^2 \\ v_{j+1} &= v_j + a_j \cdot T_{predict} \end{aligned} \tag{21}$$

The distance d_j is defined in Equation 21, where v_j is the current velocity and a_j the current acceleration. The velocity in the next prediction step v_{j+1} depends on the current acceleration a_j. In case of the current kinematic model is the *constant-velocity* model, the acceleration a_j equals zero. In case of the *constant-acceleration* model is used, the acceleration a_j received in the last CAM influences d_j and also v_{j+1}. Finally, the new position $\hat{x}_{\bar{j}}$ can be estimated based on the previous position $\hat{x}_{\bar{j}-1}$, the calculated heading θ and the driven distance d_j according to Equation 22.

$$\begin{aligned} \hat{x}_{\bar{j}-1}, \hat{x}_{\bar{j}} \in \mathbb{R}^2 \quad ; \quad \theta \in [0, 2\pi) \quad ; \quad d_j \in \mathbb{R}_0^+ \\ \hat{x}_{\bar{j}-1} = \langle x_{\bar{j}-1,lat}, x_{\bar{j}-1,long} \rangle \quad ; \quad \hat{x}_{\bar{j}} = \langle x_{\bar{j},lat}, x_{\bar{j},long} \rangle \\ x_{\bar{j},lat} = x_{\bar{j}-1,lat} + \cos(\theta) \cdot d_j \\ x_{\bar{j},long} = x_{\bar{j}-1,long} + \sin(\theta) \cdot d_j \end{aligned} \tag{22}$$

We can use this direct geometric relation, because we use cartesian coordinates within our simulations. In case of using geodetic coordinate systems, coordinates have to be converted previously.

7.2.6 *Parameter Overview*

Table 12 gives an overview about the previously introduced parameters with an according description. A more detailed description has to be given to the minimum and maximum intervals, since the annotation looks like inversion. A minimum time interval indicates the minimum time after which a new sending is allowed, i. e., the maximum sending frequency. A maximum time interval indicates the maximum time after which a new sending is required, i. e., the minimum sending frequency. In detail $T_{CPM,min}$ is the minimum passed time between two consecutive CPM broadcasts. And $T_{CPM,max}$ is the maximum time between two consecutive CPM broadcasts, which reflects the minimum CPM sending frequency.

7.3 EVALUATION

In this section we present our evaluation results. To begin with, we start with a detailed description of our simulation configuration and setup in Section 7.3.1. We compare our solution against the original Cooperative Awareness (CA) service, i. e., broadcast of CAMs, according to ETSI specification [35]. Finally, we discuss our numerical results. We show the possible reduction of channel load for position beaconing and the possible extension of vehicle perception by the use of CPMs in Section 7.3.2.

Table 12: Overview and description of the used parameters.

Parameter	Description
$\hat{x}_j$	Position vector in prediction step j.
$\hat{x}_{\bar{j}}$	Predicted position vector in prediction step j.
ε	Tolerated prediction error.
$T_{CAM,expired}$	Time interval CAM data is valid without update.
d_j	Relative distance the vehicle has moved since the last prediction step.
v_j	Vehicle velocity in prediction step j.
v_{j+1}	Vehicle velocity in the next prediction step $j+1$.
v_{j-1}	Vehicle velocity in the previous prediction step $j-1$.
a_{TH}	Threshold to switch kinematic model.
θ_j	Vehicle heading as relative angle to true north.
$CAM_{size,max}$	Maximum size of our CAM.
f_{new}	Flag, that indicates a formerly unknown vehicle.
$T_{CAM,min}$	Minimum CAM sending interval, 100ms.
$T_{CAM,max}$	Maximum CAM sending interval, 1s, 2s.
$T_{CPM,min}$	Minimum CPM sending interval, 2s.
$T_{CPM,max}$	Maximum CPM sending interval, 5s.
$T_{predict}$	Prediction step interval, 10ms.

7.3.1 *Evaluation Setup*

In the following we describe the configuration and setup of our evaluation. First, we describe the used toolset in Section 7.3.1.1, followed by a description of the used simulation scenario in Section 7.3.1.2. Afterwards, we present our simulation parameters and assumptions, as well as evaluation metrics in Section 7.3.1.3.

7.3.1.1 *Simulation Environment*

We have implemented our prototype as service into the *Artery*[1] framework. The *Artery* framework is an extension of the open source vehicular network simulation framework *Veins*[2]. *Veins* provides the interfaces for the data link communication layer and channel access, according to IEEE 802.11p standard, the de facto standard for vehicular ad-hoc communication. The *Vanetza*[3] Framework implements the ETSI *ITS-G5* communication architecture, according to the European standardization for V2X communication. It is used by *Artery* as protocol stack for the higher layers, on top of the data link layer provided by *Veins*. *Veins* is implemented in *OMNeT++*[4], that

1 https://github.com/riebl/artery
2 http://veins.car2x.org
3 https://github.com/riebl/vanetza
4 https://omnetpp.org

is the executing framework for the actual network simulation. *Veins* connects to the open traffic simulation suite SUMO[5] for the simulation of the road network, i. e., the vehicular traffic. A more detailed description of SUMO has been given in Chapter 5.3.1.1. We have used SUMO in version 0.25, OMNeT++ in version 4.6, *Veins* in version 4.4, *Vanetza* and *Artery* master branch from May 2016 to run our simulations. An overview of the single components and its interaction is depicted in Figure 33.

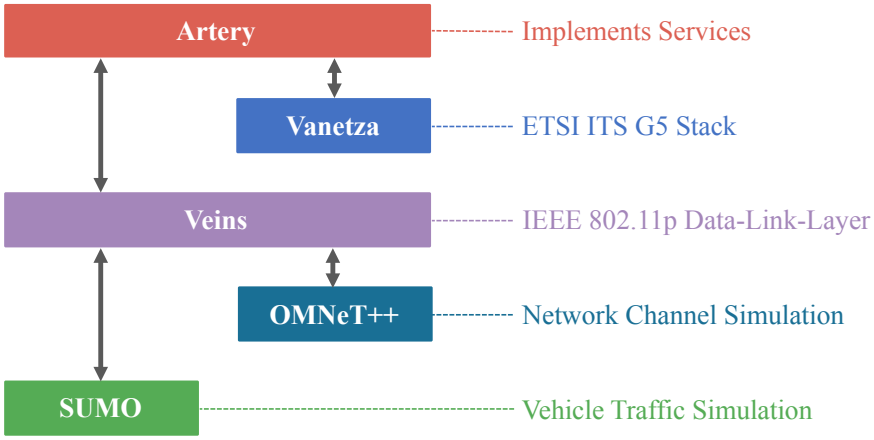

Figure 33: Overview about simulation framework architecture.

7.3.1.2 *Simulation Scenario*

As mentioned before, we have used SUMO traffic simulator for the simulation of vehicle movement. Since the same tool has been used for all simulations, details have already been introduced in Section 5.3.1. The configuration of the used road network is exactly the same as in [81, 94]. This leads to plausible comparability of our approach, with respect to channel congestion. The road network is described as a 4325m long highway with three lanes per direction. In the middle, the road is a curvy section with an S-bend of 40m radius and a length of 375m, as depicted in Figure 34. At the lane ends, these are connected by curves with the respective lane ends of the opposite lane. Thus, vehicles will not leave the simulation scenario on reaching the lane end, but turn and drive back in the opposite direction. As explained in [81], this configuration allows evaluation on straight parts as well as on a curvy section with high vehicle dynamics. This is necessary, because CAM generation also depends on vehicle dynamics. The maximum velocity of our simulated vehicles were 32m/s on the most left lane, 30m/s on the middle lane and 28m/s on the most right lane. Moreover, three vehicle types have been defined with a maximum velocity of 32m/s, 30m/s and 28m/s, respectively. These three vehicle types are dropped into the simulation randomly to the three lanes. Vehicles reduce velocity at the curves at the end of the lanes for turning. This leads to a higher vehicle density at the end of the lanes as well as in the curvy middle section. We have run our simulations with four different vehicle densities, namely 250, 500, 1000 and 1500 vehicles in parallel per scenario.

5 http://sumo.dlr.de/

Figure 34: The used road topology within our simulations, which is the same as in [81, 94].

7.3.1.3 *Simulation Parameters & Evaluation Metrics*

The model for wireless channel propagation is Nakagami distributed, as in [81, 94] and we have also used the same simulation parameters. The Nakagami distribution is a probability distribution, related to the gamma distribution, with a shape and a spread parameter. The shape parameter, and thus the signal attenuation, is in the simulation controlled by the distance of the vehicles. Hence, the packet reception probability depends on the distance of the sending and the receiving vehicle and the according probability distribution. An overview of our simulation parameters is given in Table 13. The packet reception *signal to interference plus noise ratio* (SINR) describes the minimum necessary power of the signal, divided by the sum of noise and interference power, to receive the packet.

As described in Section 4.1, the vehicular ad-hoc communication range depends on many factors. To realize a maximum communication range of 500m with the previously described transmission model, the transmission power was set to 10dBm, according to [81, 94]. Whether a new CAM has to be sent or not is checked every $T_{CAM,min} = 100ms$. For the maximum CAM sending interval $T_{CAM,max}$ we have used 1s and 2s. As values for the tolerated prediction error ε we have used 0.2m, 0.5m and 1m. We have used $0.3\frac{m}{s^2}$, $0.6\frac{m}{s^2}$ and $1\frac{m}{s^2}$ as threshold a_{TH} to switch the kinematic model. Moreover, simulations have been executed with and without transmission of CPMs. For comparison, also CAM broadcast, according to ETSI specification, has been simulated [35]. In our evaluations we have defined the time, information is treated as valid, received by CAMs, to 5s.

7.3.2 *Evaluation Results and Discussion*

In this section, we evaluate the performance of our previously introduced mechanism. We start with a comparison of our object tracking mechanism with modified CAMs and compare it against the standardized ETSI *ITS-G5* implementation in Section 7.3.2.1. Here, we show the potential reduction of channel load in terms of the amount of sent messages and CBR. In the following, we evaluate the perception extension by the use of CPM in Section 7.3.2.2. Here, we show the average as well as minimal and maximal perception extension, with respect to completeness of knowledge about moving objects in the vehicle surroundings. In general, we have used a full factorial design for our evaluations, i. e., we have evaluated all combinations of the previously introduced evaluation parameters. This results in 18 different parameter combinations plus the reference simulation with the CA service according to the ETSI *ITS-G5* standard. Each combination has been executed with four different vehicle densities as defined in Table 13, that results in total of 76 different simulation runs. The Simulation time is 360s for each. Due to the high computation time, we have set the trigger check interval for our approach to 100ms. To analyze the impact

Table 13: Simulation parameters. Wireless channel parameters according to [81, 94].

Parameter	**Value**
Noise floor	$-99\mathrm{dBm}$
Carrier sense threshold	$-96\mathrm{dBm}$
Packet SINR	$7\mathrm{dBm}$
Channel transmission rate	$6\mathrm{Mbps}$
Transmission power	$10\mathrm{dBm}$
GPS update rate	$10\mathrm{ms}$
Maximum CAM sending rate	$10\mathrm{Hz}$
Maximum CAM valid time	$5\mathrm{s}$
Number of vehicles	250, 500, 1000, 1500
Minimum CAM sending rate	$0.5\mathrm{Hz}, 1\mathrm{Hz}$
Tolerated prediction error	$0.2\mathrm{m}$, $0.5\mathrm{m}$, $1\mathrm{m}$
Threshold to switch kinematic model	$0.3\frac{\mathrm{m}}{\mathrm{s}^2}$, $0.6\frac{\mathrm{m}}{\mathrm{s}^2}$, $1\frac{\mathrm{m}}{\mathrm{s}^2}$

of a shorter trigger check interval, we have executed the evaluation runs also with a check interval of $10\mathrm{ms}$ for the scenario of 500 vehicles. The influence of a higher trigger check rate will be discussed in Section 7.3.2.3. In total, execution of these simulations has last more than 2000 hours on a server with two Intel Xeon E5-2643v3 6-Core CPUs, with up to $3,4\ \mathrm{GHz}$ per core.

7.3.2.1 *Reduction of Communication Costs*

By the use of our previously introduced CAM sending mechanism, we are able to reduce channel load for object tracking. We have simulated the scenario described in Section 7.3.1.2 with 250, 500, 1000 and 1500 vehicles. These vehicles were placed on the road during the whole simulation. Thus, the respective number of vehicles describes the amount of vehicles within the simulation scenario at anytime for the complete run. We analyze performance improvement of our object tracking mechanism in comparison to the standardized ETSI *ITS-G5* Cooperative Awareness (CA) service in terms of channel load. For evaluation, we have used a full factorial design, that results in 18 run configurations r.1 to r.18, for each vehicle density of 250, 500, 1000 and 1500 vehicles. In addition, we have simulated the reference run r.0 with the CAM sending mechanism according to standardized ETSI *ITS-G5* implementation with DCC, also for the four different vehicle densities. An overview of the evaluation run configurations r.0 to r.18 is given in Table 14.

Total number of messages: To give an impression of the overall performance of our object tracking mechanism, we start with a comparison of the amount of generated CA messages over time. Figure 35 shows the accumulated CAMs in the scenario with 1000 vehicles. Each, for r.0 with the CAM sending mechanism, according to standardized ETSI *ITS-G5* implementation with DCC, and r.1 and r.10 for our approach as comparison. We have selected r.0 and r.10, with $T_{CAM,min}$ of 0.5 Hz and 1 Hz as an example configuration of our approach, since the minimum CAM sending

Table 14: Run configurations of the used parameters.

Run configuration	Minimum CAM sending rate	Tolerated prediction error	Threshold to switch kinematic model
r.0	CAM according to standardized *ETSI ITS-G5* implementation with DCC.		
r.1	1Hz	0.2m	$0.3\frac{m}{s^2}$
r.2	1Hz	0.2m	$0.6\frac{m}{s^2}$
r.3	1Hz	0.2m	$1.0\frac{m}{s^2}$
r.4	1Hz	0.5m	$0.3\frac{m}{s^2}$
r.5	1Hz	0.5m	$0.6\frac{m}{s^2}$
r.6	1Hz	0.5m	$1.0\frac{m}{s^2}$
r.7	1Hz	1.0m	$0.3\frac{m}{s^2}$
r.8	1Hz	1.0m	$0.6\frac{m}{s^2}$
r.9	1Hz	1.0m	$1.0\frac{m}{s^2}$
r.10	0.5Hz	0.2m	$0.3\frac{m}{s^2}$
r.11	0.5Hz	0.2m	$0.6\frac{m}{s^2}$
r.12	0.5Hz	0.2m	$1.0\frac{m}{s^2}$
r.13	0.5Hz	0.5m	$0.3\frac{m}{s^2}$
r.14	0.5Hz	0.5m	$0.6\frac{m}{s^2}$
r.15	0.5Hz	0.5m	$1.0\frac{m}{s^2}$
r.16	0.5Hz	1.0m	$0.3\frac{m}{s^2}$
r.17	0.5Hz	1.0m	$0.6\frac{m}{s^2}$
r.18	0.5Hz	1.0m	$1.0\frac{m}{s^2}$

rate has the strongest impact, according to the amount of sent messages. If a CAM is broadcasted depends on several conditions. The mechanism for our approach has been explained in Section 7.2. The CA service mechanism, according to ETSI standard has been explained in Section 2.4.2. If a new CAM according to ETSI standard will be broadcasted, also depends on the DCC mechanism that has been explained in Section 2.4.2. Thus, the CAM sending rate according to ETSI standard also depends on channel load. Both, ETSI standard and our approach, have thresholds according to a change in position information. For the example in Figure 35, the CAM sending rate, according to ETSI standard, results in approximately 5 Hz and accumulates over the simulation time of 360s to a total of 1769 CAM messages. Thus, we can see that a high channel load caused a reduction of CAM broadcasts. With our approach in configuration r.1 we need in the same setting only a total of 351 CAM messages and in configuration r.10 only a total of 211 messages. This shows a reduction of CA messages of up to 90%, which results in a much lower channel busy rate. This is the prerequisite to broadcast perception information, to allow neighbor vehicles to extend their local perception, which is analyzed in detail in Section 7.3.2.2. After this introducing example, we will deeper analyze the behavior of our object tracking approach in comparison to the standardized ETSI *ITS-G5* CA service in the following.

Average number of messages: In the next step, we compare the accumulated average number of CA messages for the four vehicle densities of 250, 500, 1000 and 1500

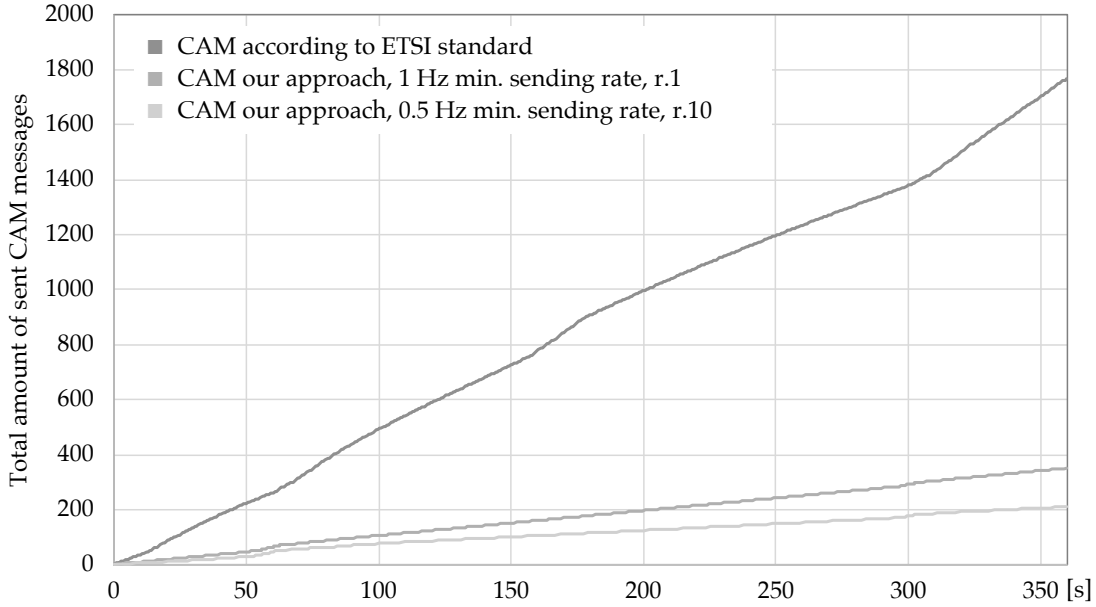

Figure 35: Example illustration of the number of generated CAM messages over time for one example vehicle. Comparison of standardized ETSI *ITS-G5* CAM implementation to our approach in a scenario with 1000 vehicles. For our approach we have selected run configuration r.1 and r.10 (cf. Table 14), which have a minimum CAM sending rate of 0.5 Hz and 1 Hz, each with a tolerated prediction error of 0.2m and a threshold, to switch kinematic model, of $0.3\frac{m}{s^2}$.

vehicles for the whole simulation time of 360s. We compare the standardized ETSI *ITS-G5* CAM implementation to all different evaluation run configurations r.1 to r.18 of our approach, as listed in Table 14. The results, with respect to the mean of total CAMs sent, are given in Figure 36. In addition, the figure denotes the respective confidence intervals at a confidence level of 95% (i.e., $\alpha = 0.05$). Confidence intervals were computed using Student's t-distribution [154]. Since the confidence intervals are very small, these are poorly visible in Figure 36. Therefore, we provide the evaluation result data in Tables 28, 29, 30 and 31 in Section A.3.3. It can be clearly seen from Figure 36, that the average number of generated CA messages per vehicle in the reference implementation, i. e., standardized ETSI *ITS-G5* CAM implementation executed in r.0, strongly decreases with an increasing number of vehicles, i. e., a higher vehicle density. As a result, the object tracking accuracy decreases if no trajectory prediction would be incorporated. In contrast to this, our approach shows a strongly constant behavior, with respect to the amount of sent CA messages. Moreover, our approach needs up to 90% less messages per vehicle on average. This results in a much lower channel busy rate, that will be discussed in the following. One can see a drop in the average amount of CA messages between run configurations r.1 to r.9 and r.10 to r.18 of about 45%. This is due to the most influencing parameter $T_{CAM,max}$ of the maximum CAM sending interval, which corresponds to the minimum CAM sending rate. For r.1 to r.9, the minimum CAM sending rate is 1 Hz and for r.10 to r.18 it is 0.5 Hz. The tolerated prediction error and the threshold to switch the kinematic model for r.1 to r.9 have effectively no influence, since $T_{CAM,max}$ mostly causes the CAM sending trigger. For r.10 to r.18 we can see some influence of the tolerated prediction error and the threshold to switch the kinematic model of about 9%. The

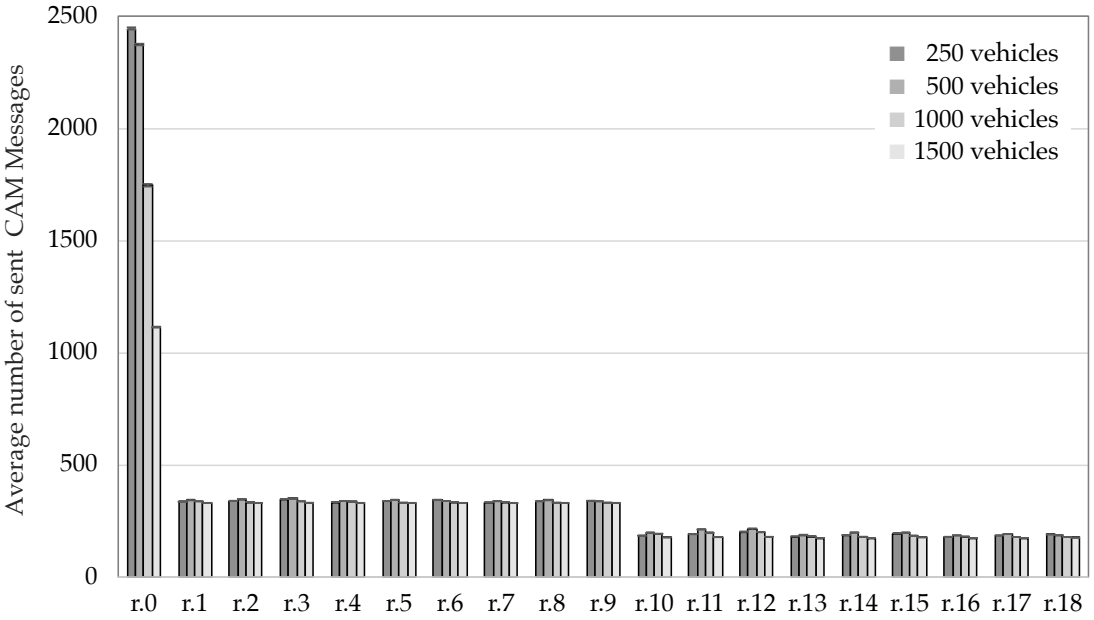

Figure 36: Average number of generated CAM messages per vehicle. Comparison of standardized ETSI *ITS-G5* CAM implementation to our approach for all different evaluation run configurations r.1 to r.18, as described in Section 7.3.1.3 and listed in Table 14. Each for the scenario with 250, 500, 1000 and 1500 vehicles.

lowest average amount of sent CA messages is for configuration r.16, caused by the highest tolerated prediction error. A slightly decreasing number of sent messages, according to the vehicle density, is caused by a reduced dynamic of the vehicles, due to congestion. As examples we will have a closer look into the scenarios with 1000 and 1500 vehicles. In the reference implementation, run r.0, we have an average of

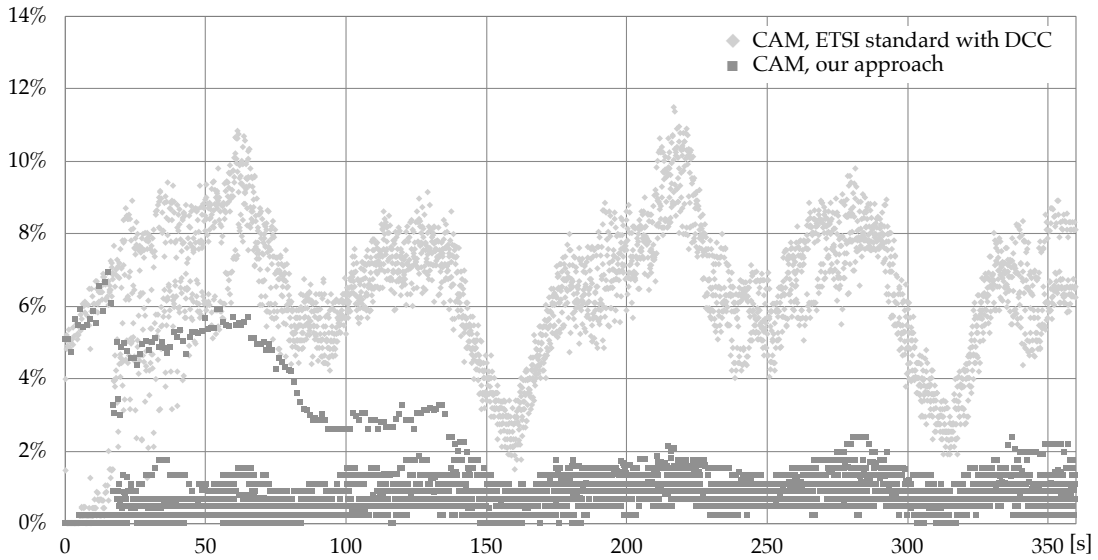

Figure 37: Example illustration of Channel Busy Ratio (CBR) over time for one example vehicle in a scenario with 250 vehicles. In the depicted example, the minimum CAM sending rate is set to 0.5 Hz, the tolerated prediction error to 0.2m and the threshold to switch the kinematic model to $0.3\frac{m}{s^2}$.

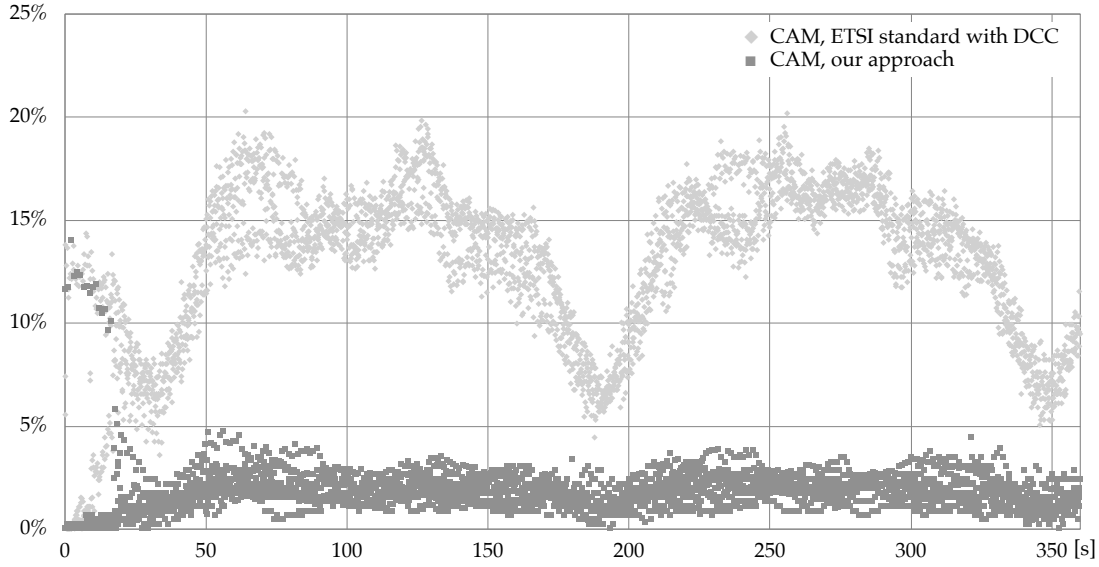

Figure 38: Example illustration of Channel Busy Ratio (CBR) over time for one example vehicle in a scenario with 500 vehicles. In the depicted example, the minimum CAM sending rate is set to 0.5 Hz, the tolerated prediction error to 0.2m and the threshold to switch the kinematic model to $0.3\frac{m}{s^2}$.

about 1746 messages per vehicle for the scenario with 1000 vehicles and about 1114 messages per vehicle for the scenario with 1500 vehicles. When comparing with r.1, we achieve a reduction of the average amount of message of about 81% for the scenario with 1000 vehicles and about 70% for the scenario with 1500 vehicles. If we use for comparison r.10, we achieve a reduction of the average amount of message of about 89% for the scenario with 1000 vehicles and about 84% for the scenario with 1500 vehicles. This results for the reference implementation, run r.0, for both scenarios in a total of about 1.7 million messages, which results in about 1180 messages per second and road kilometer. With our approach, we can reduce this for r.1 to about 234 for the scenario with 1000 vehicles and to about 343 for the scenario with 1500 vehicles. For run configuration r.10, we can reduce this to about 134 for the scenario with 1000 vehicles and to about 183 for the scenario with 1500 vehicles. We achieved the best result in run configuration r.17, with about 123 messages per second and road kilometer, for the scenario with 1000 vehicles. In the scenario with 1500 vehicles we can see almost similar results for all nine run configurations.

Channel busy rate analysis: As mentioned before, the amount of sent messages strongly influences the channel busy rate. Thus, we will have a closer look into this in the following. The CBR is defined as the fraction of the time the channel is busy over the observation time. We have to mention, that our collected CBR values are always average values over 100ms. Thus, there might be fluctuations within these 100ms and absolute values in between these 100ms must be higher. This can be seen on the average CA message rate of r.0, which is controlled accordingly to Table 11. We can derive from Figure 36 and Tables 30 and 31 an average CAM transmission rate of about 5 Hz for the scenario with 1000 vehicles and of about 3 Hz for the scenario with 1500 vehicles. In the following, an illustration of the CBR over time for one example vehicle in the four scenarios with 250, 500, 1000 and 1500 vehicles, is shown in Figures 37, 38, 39 and 40. The figures show a comparison of the reference

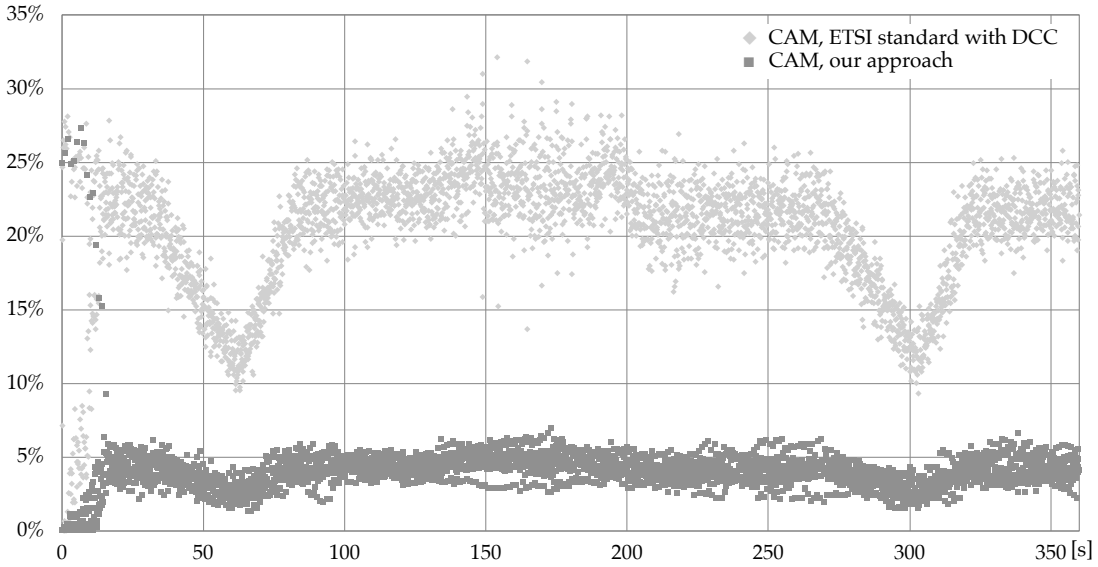

Figure 39: Example illustration of Channel Busy Ratio (CBR) over time for one example vehicle in a scenario with 1000 vehicles. In the depicted example, the minimum CAM sending rate is set to 0.5 Hz, the tolerated prediction error to 0.2m and the threshold to switch the kinematic model to $0.3\frac{m}{s^2}$.

implementation, run r.0, and our approach in run configuration r.10. Here, we have to mention that in the reference implementation the transmission trigger and CBR is checked every 10ms, whereas in our implementation we have used an interval of 100ms, due to computation time. However, an increase of the trigger check rate in our approach might further decrease CBR and PER. For comparison, we have executed the evaluation runs also with a check interval of 10ms for the scenario of 500 vehicles. The influence of a higher trigger check rate will be discussed in Section 7.3.2.3. We can see in Figure 37, that our approach almost constantly stays below a CBR of 2%. Whereas for r.0 the CBR considerably stronger fluctuates and goes up to values of about 11%. If we increase the number of vehicles to 500, as depicted in Figure 38, we can see our approach still in the range of $2-3\%$. For r.0 the CBR goes up to values of about 20%, fluctuates strongly and stays always above 5%. For a further increase of the number of vehicles to 1000, as depicted in Figure 39, we can see our approach around 5% with a maximum of about 6.8%. The first seconds are not representative, due to congestion on vehicle placement within the scenario. For r.0 in comparison, the CBR goes up to values of about 30%. Finally, in the scenario with the highest vehicle density, i. e., 1500 vehicles in total, we can see a further increase in CBR as depicted in Figure 40. While the CBR is relatively constant around $5-8\%$ for our approach, the CBR for r.0 goes up to about 32% and is on average much higher than for lower vehicle densities.

Figures 38 to 40 give an impression of the behavior of our approach in comparison to the ETSI CA service reference implementation in r.0, with respect to CBR over time. To analyze the overall behavior, we give the mean of the average CBR for r.0 to r.18 for the four used densities of 250, 500, 1000 and 1500 vehicles in Figure 41. In addition, the figure denotes the respective confidence intervals at a confidence level of 95% (i.e., $\alpha = 0.05$). Confidence intervals were computed using Student's t-distribution [154]. Since the confidence intervals are very small, these are poorly

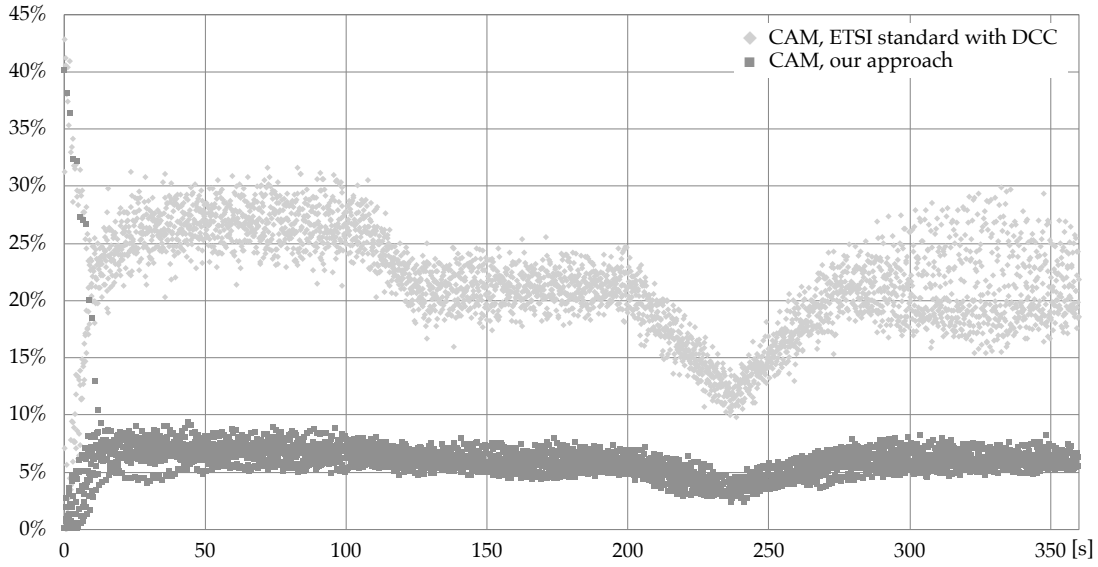

Figure 40: Example illustration of Channel Busy Ratio (CBR) over time for one example vehicle in a scenario with 1500 vehicles. In the depicted example, the minimum CAM sending rate is set to 0.5 Hz, the tolerated prediction error to 0.2m and the threshold to switch the kinematic model to $0.3\frac{m}{s^2}$.

visible in Figure 41. Therefore, we provide the evaluation result data in Tables 32, 33, 34 and 35 in Section A.3.3. We can see for all run configurations a clear increase in CBR. For our approach, i. e., r.1 to r.18, it shows a relatively linear increase. For r.0 we also see an increase, with an increasing vehicle density from 250 to 1000 vehicles, but for 1500 vehicles the value remains roughly the same as for 1000 vehicles. This

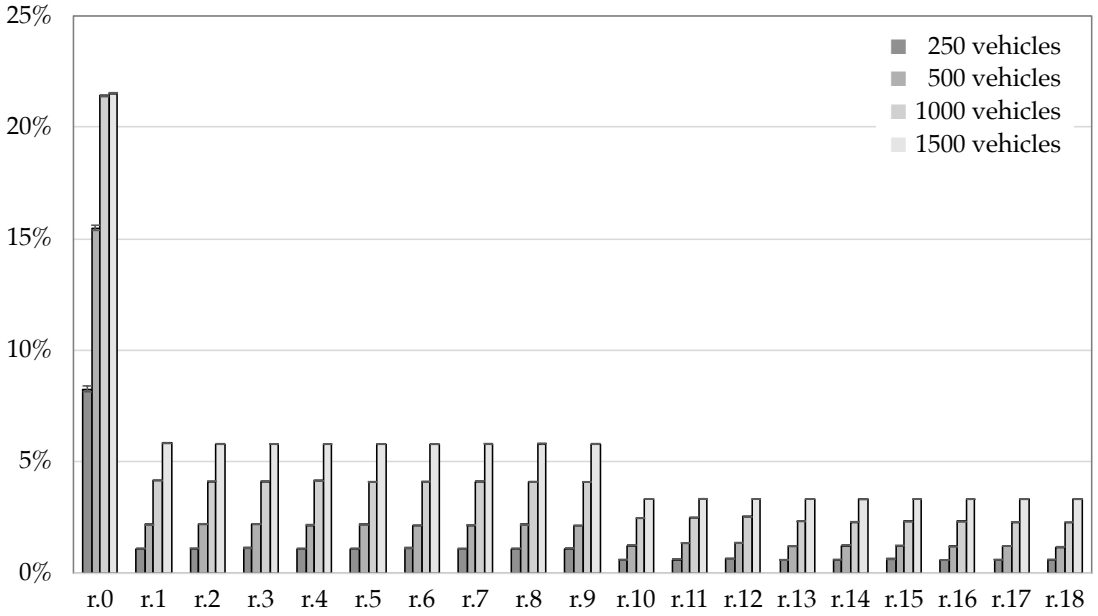

Figure 41: Average Channel Busy Ratio (CBR). Comparison of standardized ETSI *ITS-G5* CAM implementation with our approach for all different evaluation run configurations r.1 to r.18, as described in Section 7.3.1.3 and listed in Table 14. Each, for the scenario with 250, 500, 1000 and 1500 vehicles.

Table 15: CAM sending trigger conditions.

Condition	Description
Trigger 1	A change in the used kinematic model K, i.e., constant velocity or constant acceleration model.
Trigger 2	The time that has passed since the last broadcast of a CAM has exceeded $T_{CAM,max}$, the minimal CAM sending rate.
Trigger 3	A lane or road change is detected.
Trigger 4	The error in prediction exceeds the threshold ε.

shows, that for an average CBR of about 21.5% the system achieves a steady state. This is caused by the fact, that with a higher vehicle density, also the average vehicle velocity decreases, due to congestion, which also reduces the amount of CAM send triggers. However, the main reason for a saturation of CBR is that 802.11p has a decreasing throughput for an increasing amount of participants [159].

Analysis of the packet error rate: The next performance indicator we want to analyze is the Packet Error Rate (PER). The PER is defined as the number of received packets with at least one bit incorrect, divided by the total amount of received packets. The average PER for all run configurations and all four vehicle densities is given in Figure 42. Here, the ETSI standard executed in r.0 outperforms our approach. However, the bad PER results are mainly caused by the large trigger check interval of 100ms and the fact, that all vehicles are dropped to the simulation roughly simultaneously. A

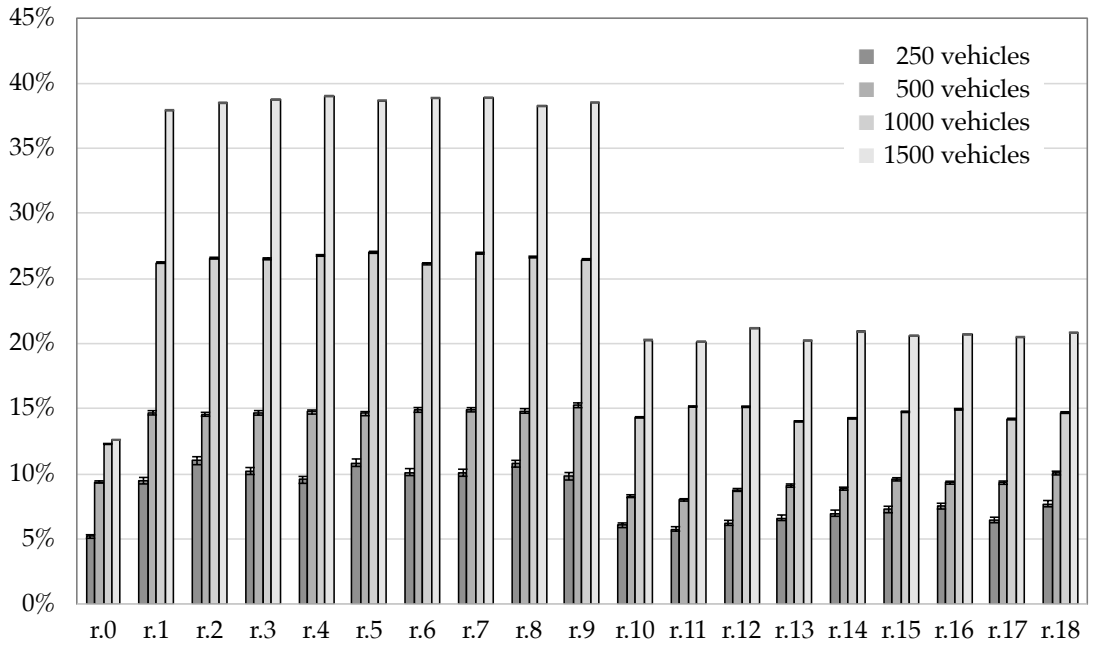

Figure 42: Average Packet Error Rate (PER). Comparison of standardized ETSI *ITS-G5* CAM implementation with our approach for all different evaluation run configurations r.1 to r.18, as described in Section 7.3.1.3 and listed in Table 14. Each, for the scenario with 250, 500, 1000 and 1500 vehicles.

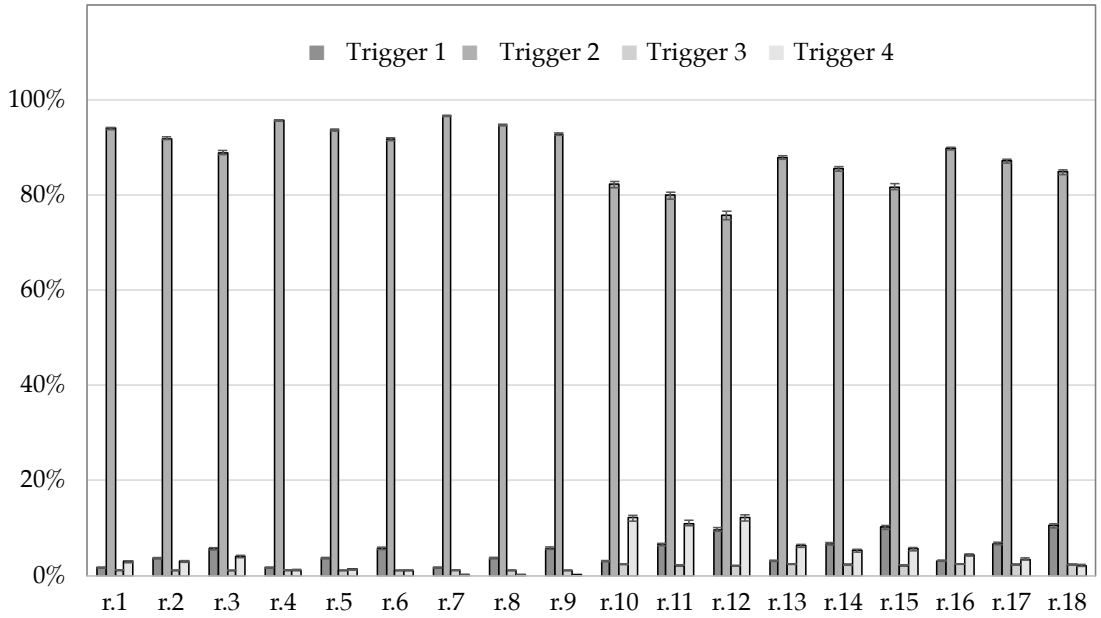

Figure 43: Average trigger condition to send a new CAM in a scenario with 250 vehicles for all different evaluation run configurations r.1 to r.18, as described in Section 7.3.1.3 and listed in Table 14.

higher trigger check rate will most probably decrease the PER and as mentioned before, we will discuss this with some examples in more detail in Section 7.3.2.3. Nevertheless, we can see a relatively linear increase in PER in relation to the amount of messages, caused by the vehicle density. Due to this, the PER is not significantly

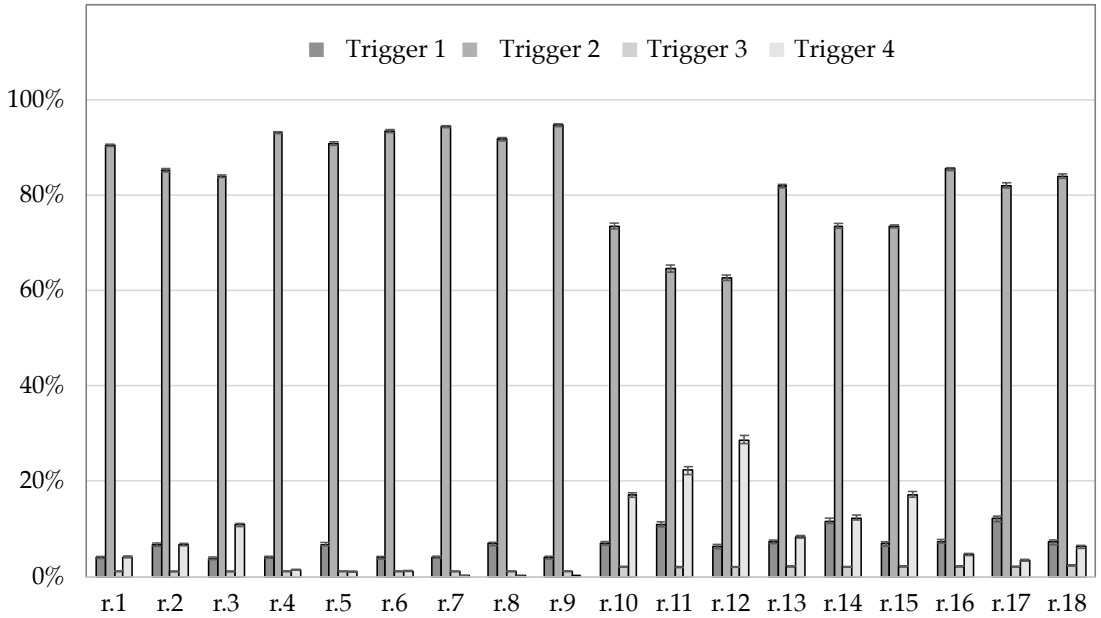

Figure 44: Average trigger condition to send a new CAM in a scenario with 500 vehicles for all different evaluation run configurations r.1 to r.18, as described in Section 7.3.1.3 and listed in Table 14.

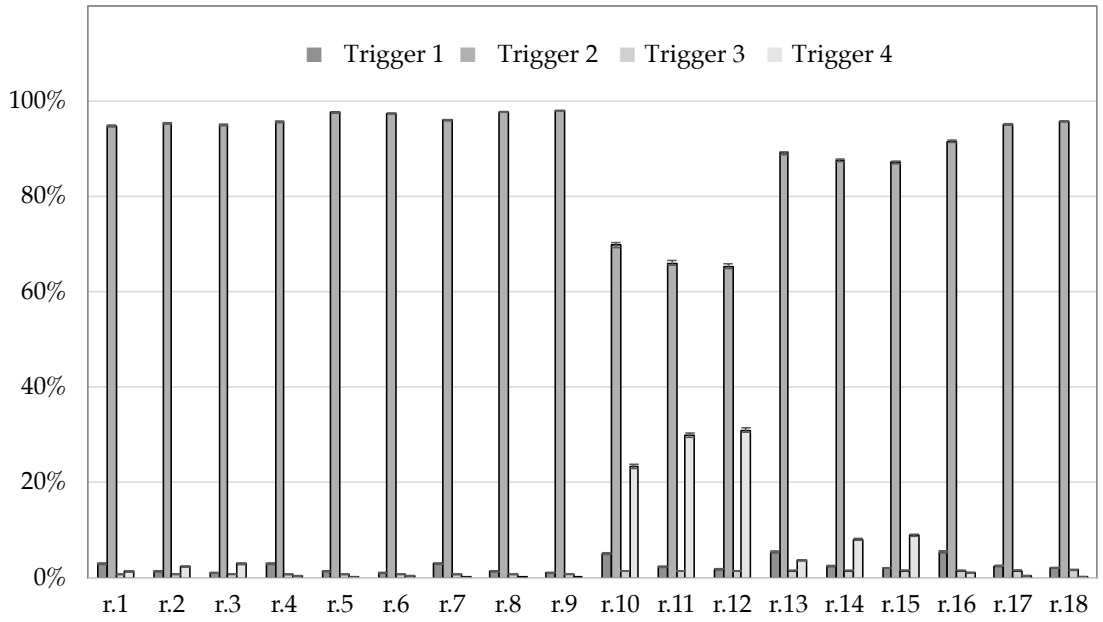

Figure 45: Average trigger condition to send a new CAM in a scenario with 1000 vehicles for all different evaluation run configurations r.1 to r.18, as described in Section 7.3.1.3 and listed in Table 14.

increasing for r.0 from the scenario with 1000 vehicles to the scenario with 1500 vehicles, since also the amount of messages remains similar.

Analysis of the send trigger: To understand how the different parameters influence our approach, we will analyze the reason to send out a new CAM in the following. If a new CAM will be sent, depends on four conditions, as described in Section 7.2.1. We have summarized these trigger conditions in Table 15. If one of these conditions is triggered, a new CAM will be broadcasted. We have evaluated how often each of these conditions is triggered for all 18 different evaluation configurations r.1 to r.18, as described in Section 7.3.1.3 and for the four different vehicle densities of 250, 500, 1000 and 1500 vehicles in our scenario. This gives an indication of the performance of our approach and possible reductions of the CAM sending rate. The respective results are depicted in Figures 43, 44, 45 and 46. Figure 43 gives the results for the scenario of 250 vehicles. Here, trigger 2, the minimal CAM sending rate, strongly dominates. This means, that the passed time since the last broadcast of a CAM, has exceeded $T_{CAM,max}$. However, for an increased $T_{CAM,max}$ of 2 seconds in r.10 to r.18, we can also see the other triggers. For r.10, r.11 and r.12, we can see a value of about 12% for trigger 4, that indicates an error in prediction. For these three configurations, we have the smallest tolerated prediction error ε of 0.2m. For r.15 and r.18, we can see a value of about 8% for trigger 1, that indicates a change in the use of the kinematic model. This is only slightly higher than for the other run configurations. For the higher vehicle density of 500 vehicles, as depicted in Figure 44, we can see the result of much higher vehicle dynamics. The rate of trigger 4, that indicates an error in prediction, goes up to about 29% for r.12 and is also relatively high for r.3 and r.10 to r.15. This might be caused by the concertina effect, if the vehicle density is relatively high, but low enough to allow high dynamics. If we further increase the vehicle density to 1000, as depicted in Figure 45, we can see an interesting effect: Now the vehicle dynamics seem to slow down a bit. Only for

r10 to r.12, that have the smallest tolerated prediction error ε of 0.2m, trigger 4 has large values of up to 31%. For all other run configurations, trigger 2, i. e., the time, is very dominant. This effect is enhanced for the next higher vehicle density of 1500, as depicted in Figure 46. Here, we can see quasi solely trigger 2. As expected, for the run configuration with the largest freedom, namely r.16, condition 2 is mostly triggered overall.

In summary we can see from our results, presented in Figures 35 to 46, that we are able to significantly reduce the channel load for object tracking. This reduction can be used for the extension of the vehicle perception by the use of CPMs, that we will analyze in the following.

7.3.2.2 *Extension of Vehicular Perception*

The basic concept to reduce the channel load for moving object tracking, as analyzed in Section 7.3.2.1, is to get enough channel capacity for forwarding information of moving objects. As explained in Chapter 4.3, many use cases need information about moving objects in a distance beyond single hop ad-hoc communication range. Our CPM contains information of up to 27 known neighbor vehicles. As mentioned before, we select the 27 known vehicles in the largest distance to the ego vehicle. Without any information forwarding, the perception range is about 500m in our simulation setup, since this is the configured single hop communication range. In the first step, we analyze the average maximum perception range, which is defined as the average distance of the farthest known vehicle. In Figure 47, the respective simulation results are depicted. Since the confidence intervals are very small, these are poorly visible in Figure 47. Therefore, we provide the evaluation result data in Table 41 in Section A.3.3. It can be clearly seen, that for all run configurations and all simulated vehicle densities, the maximum perception range is much higher than

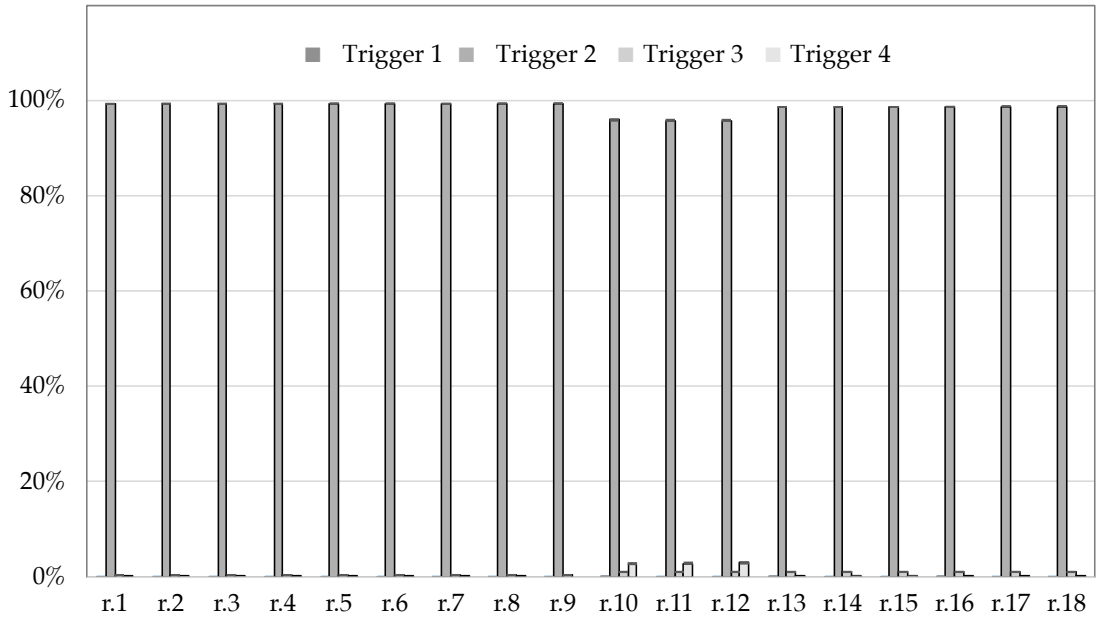

Figure 46: Average trigger condition to send a new CAM in a scenario with 1500 vehicles for all different evaluation run configurations r.1 to r.18, as described in Section 7.3.1.3 and listed in Table 14.

the one hop ad-hoc communication range. In case of a vehicle density of 250 vehicles in the scenario, the average maximum perception range is in the range of about $1203 - 1298$m for all run configurations. In case of a minimum CAM sending rate of 1 Hz, i. e., r.1 – r.9, the average maximum perception range is about 1500m for a vehicle density of 500 and 1000 vehicles. For a vehicle density of 1500 vehicles, the average maximum perception range drops down to about 1300m. This is caused by a high channel load, caused by the propagation of CAMs, due to the high number of vehicles. For a lower CAM sending rate of 0.5 Hz, i. e., $r.10 - r.18$, the average maximum perception range is a lot higher. The highest value is about 2154m and can be seen for r.15 in the scenario of 1000 vehicles. Also for $r.10 - r.18$, the average maximum perception range drops down, due to a higher channel usage for CAM propagation in the scenario of 1500 vehicles. However, values of the average maximum perception range remain in the range of $1750 - 1855$m. These results reflect the fact that in case of a higher vehicle density not only the wireless channel gets more crowded for CAM propagation, but also vehicles know much more neighbors and only a subset of these can be forwarded as CPMs.

This effect is getting even clearer in our following results about the completeness of knowledge. We define the average completeness of local vehicle knowledge as the ratio of known vehicles to existing vehicles, in the respective distance. We have analyzed this completeness for the distances of 250m, 500m, 750m and 1000m. The respective results are depicted in Figures 48, 49, 50 and 51. Since the confidence intervals are very small, these are poorly visible. Therefore, we provide the respective evaluation result data in Tables 42, 43, 44 and 45 in Section A.3.3. Figure 48 gives the results for the scenario of 250 vehicles. The relatively low vehicle density in this scenario causes very homogenous behavior across all run configurations. In the distances of 250m, 500m and 750m, the completeness is for all run configurations very

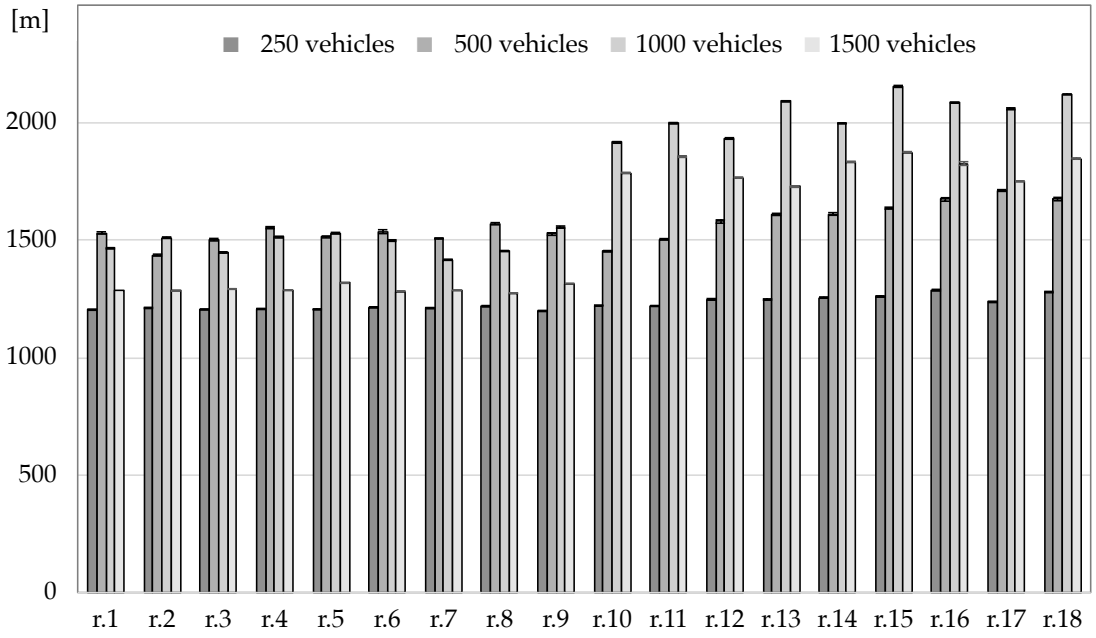

Figure 47: Average of the maximum perception range for all different evaluation run configurations r.1 to r.18, as described in Section 7.3.1.3 and listed in Table 14. Each, for the scenario with 250, 500, 1000 and 1500 vehicles.

high, with values in the range of $96-99\%$. In a distance of 1000m, the completeness drops down to about 85%, which can also be explained by the relatively low vehicle density. This changes in case the scenario with 500 vehicles is used, as depicted in Figure 49. The vehicle density in this scenario is still relatively low, i. e., the wireless channel is not crowded, because of CAM propagation. This causes a very homogenous behavior across all run configurations. In the distances of 250m, 500m and 750m, the completeness is for all run configurations very high, with values in the range of $96-98\%$.

In a distance of 1000m, the completeness drops down to about $88-90\%$, which can also be explained by the relatively low vehicle density. However, these are very high values that reflect a nearly complete knowledge about moving neighbor vehicles, even in 1000m distance. Figure 50 shows the result values for a scenario of 1000 vehicles. For run configurations r.1 – r.9, the behavior is very similar with even better results in 1000m distance. The average completeness of local vehicle knowledge is for the 1000m distance in the range of 93%. For run configurations r.10 – r.18, the average completeness decreases. This is caused by two factors: First, the CAM sending rate is only 0.5 Hz for r.10 – r.18 and thus, the completeness of knowledge about vehicles in the direct communication range decreases. Second, we can see for r.10 – r.18 an increased perception range, which causes a lower level of completeness. Similar results can be seen for the scenario of 1500 vehicles, as shown in Figure 51. The higher vehicle density causes a higher channel load, due to CAM propagation, which increases the previously mentioned effects. For r.1 – r.9 the average completeness of local vehicle knowledge decreases to about 90% in 1000m distance. In case of r.10 – r.18, the average completeness of local vehicle knowledge decreases to about 77%. Thus, the effects of a higher channel load are clearly visible.

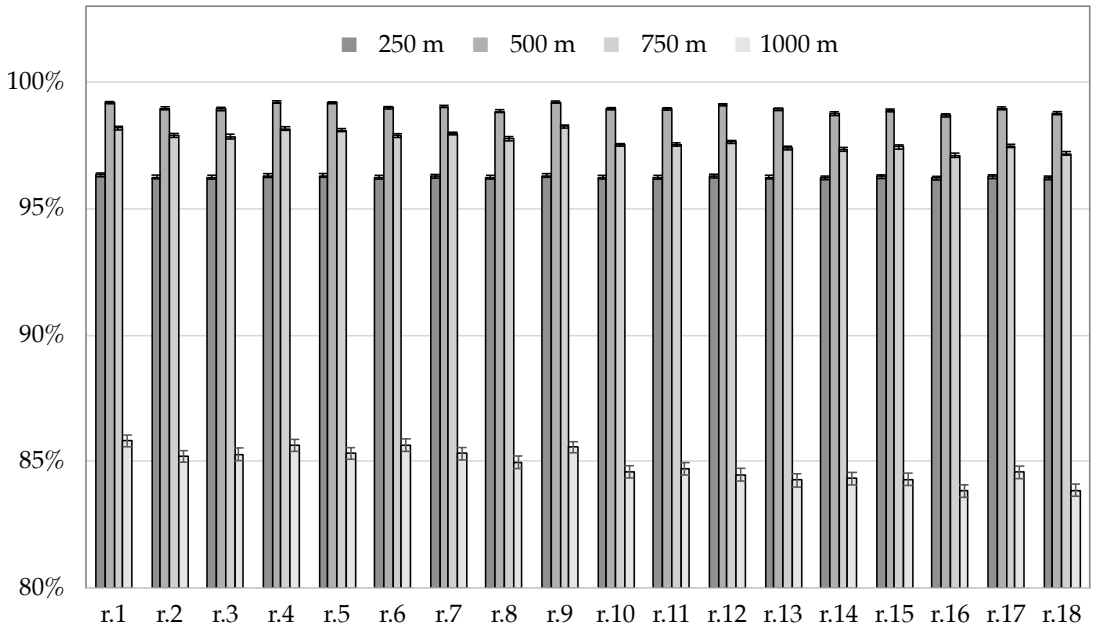

Figure 48: Average completeness of local vehicle knowledge, with respect to the ratio of known vehicles to existing vehicles in the respective distance. The figure gives results for a scenario with 250 vehicles for all different evaluation run configurations r.1 to r.18, as described in Section 7.3.1.3 and listed in Table 14.

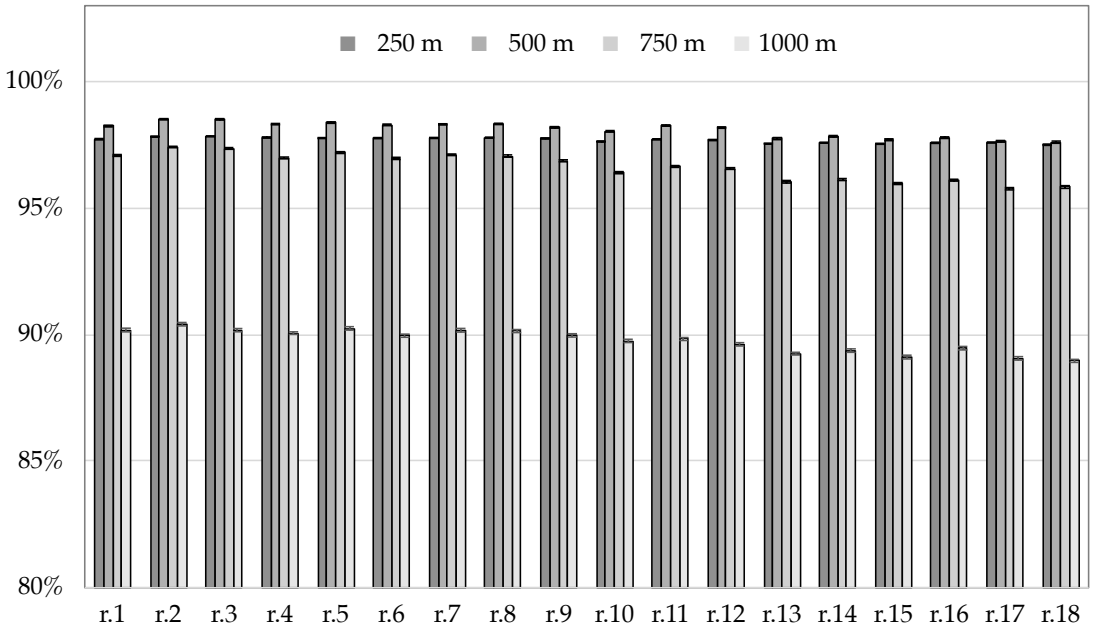

Figure 49: Average completeness of local vehicle knowledge, with respect to the ratio of known vehicles to existing vehicles in the respective distance. The figure gives results for a scenario with 500 vehicles for all different evaluation run configurations r.1 to r.18, as described in Section 7.3.1.3 and listed in Table 14.

Analysis of the channel busy rate: In the next step we analyze the effect of our CAM forwarding mechanism, i. e., broadcasting CPMs, to the channel load. First, an illustration of the CBR over time for one example vehicle in the four scenarios of

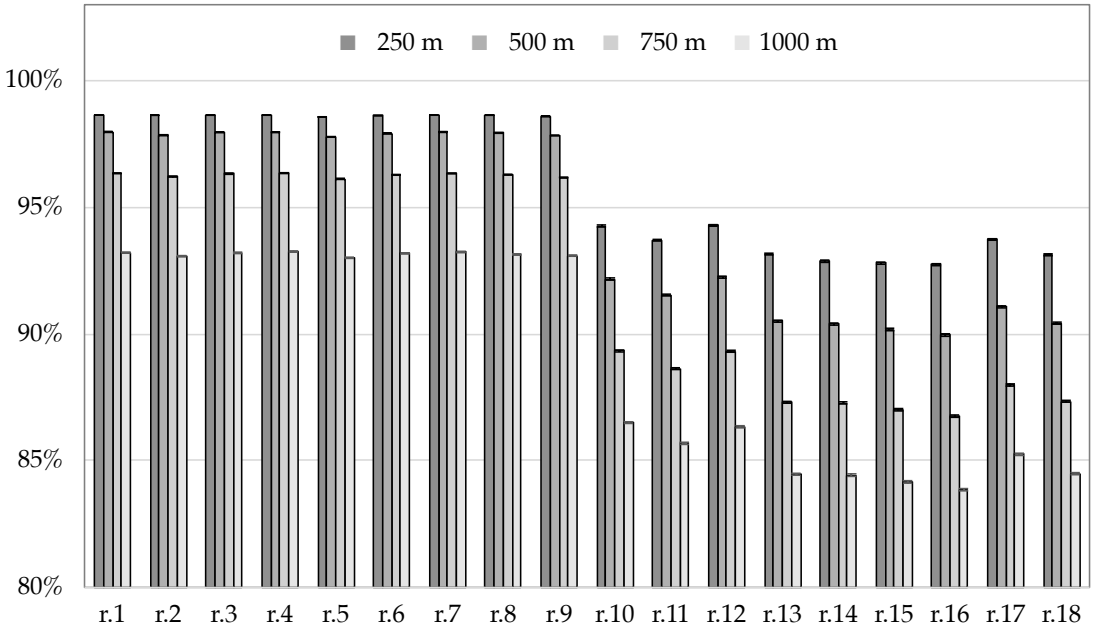

Figure 50: Average completeness of local vehicle knowledge, with respect to the ratio of known vehicles to existing vehicles in the respective distance. The figure gives results for a scenario with 1000 vehicles for all different evaluation run configurations r.1 to r.18, as described in Section 7.3.1.3 and listed in Table 14.

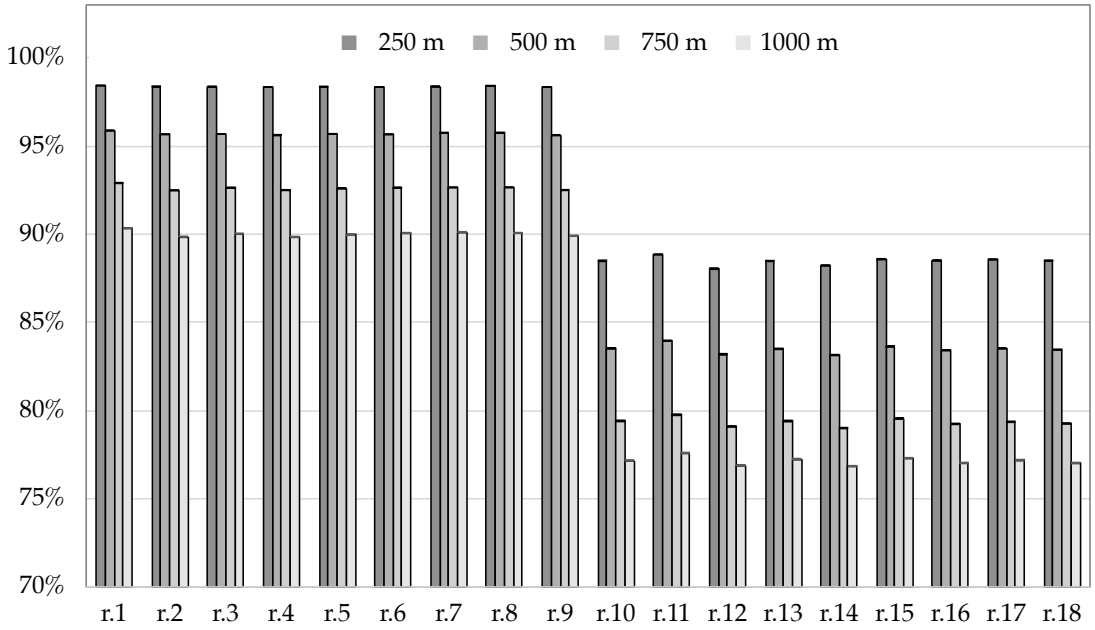

Figure 51: Average completeness of local vehicle knowledge, with respect to the ratio of known vehicles to existing vehicles in the respective distance. The figure gives results for a scenario with 1500 vehicles for all different evaluation run configurations r.1 to r.18, as described in Section 7.3.1.3 and listed in Table 14.

250, 500, 1000 and 1500 vehicles is shown in Figures 52, 53, 54 and 55. The figures show a comparison of the ETSI *ITS-G5* CAM reference implementation, i. e., run r.0, and our approach in run configuration r.10, i. e., a minimum CAM sending rate of 0.5 Hz, a tolerated prediction error of 0.2m and a threshold to switch the kinematic model of $0.3\frac{m}{s^2}$. Similar, as mentioned before, in the ETSI *ITS-G5* CAM reference im-

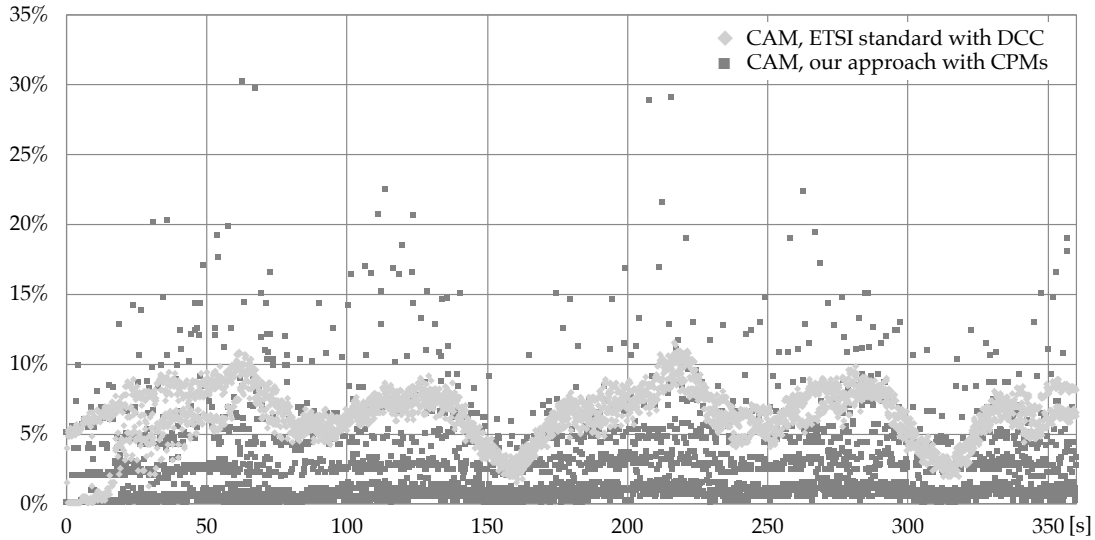

Figure 52: Example illustration of Channel Busy Ratio (CBR) with the use of CPMs over time for one example vehicle. The figure gives results for a scenario with 250 vehicles. For our approach, the example shows run configuration r.10, i. e., a minimum CAM sending rate of 0.5 Hz, a tolerated prediction error of 0.2m and a threshold to switch the kinematic model of $0.3\frac{m}{s^2}$.

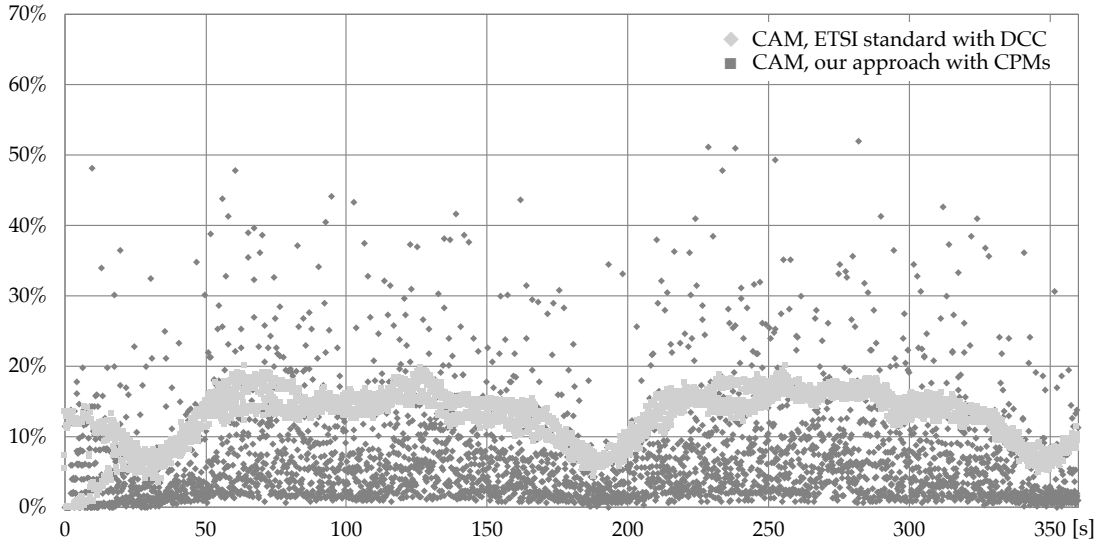

Figure 53: Example illustration of Channel Busy Ratio (CBR) with the use of CPMs over time for one example vehicle. The figure gives results for a scenario with 500 vehicles. For our approach, the example shows run configuration r.10, i. e., a minimum CAM sending rate of 0.5 Hz, a tolerated prediction error of 0.2m and a threshold to switch the kinematic model of $0.3\frac{m}{s^2}$.

plementation, the transmission trigger and CBR is checked every 10ms, whereas in our implementation we have used an interval of 100ms, due to computation time. However, an increase of the trigger check rate in our approach, might further decrease CBR and PER. To show this effect, we have executed the evaluation for all run configurations, also with a check interval of 10ms for the scenario of 500 vehicles. The influence of a higher trigger check rate will be discussed in Section 7.3.2.3 and the respective results for CBR are depicted in Figure 63. For the lowest vehicle density in the scenario of 250 vehicles, as depicted in Figure 52, the CBR is in general relatively low at values below 10%. Our approach mostly remains below 5% CBR, only some data points jump up to values of up to 30%. Overall the figure shows that our approach has mostly lower CBR values for CAM and CPM together, compared to the ETSI *ITS-G5* CAM reference implementation, for low vehicle densities. Similar results can be seen for the next higher vehicle density in the scenario of 500 vehicles, as depicted in Figure 53. The CBR increases for the ETSI *ITS-G5* CAM reference implementation, mostly to values between 10% and 20%. The figure shows that our approach overall remains below these values, mostly even below 10%. Only some data points jump up to values of up to 40% and some few up to 50%. The following Figure 54 gives the results for the next higher vehicle density in the scenario of 1000 vehicles. The CBR increases for the ETSI *ITS-G5* CAM reference implementation, mostly to values between 20% and almost 30%. The two minima at about 60s and 300s are caused by the straight elements of our simulation road network. The result values of our approach have a wider distribution than for the scenarios of 250 and 500 vehicles and slightly follow the distribution of the values of the ETSI *ITS-G5* CAM reference implementation. Overall, the figure shows that our approach has still mostly lower CBR values for CAM and CPM together, compared to the ETSI *ITS-G5* CAM reference implementation. But several result values are above these values in the range up to 40%. Only some data points jump up to values of up to 50% and

some few up to 60%. For the highest examined vehicle density in the scenario of 1500 vehicles, as depicted in Figure 55, the result value distribution of our approach is significantly wider and has higher values. A large proportion follows the values of the ETSI *ITS-G5* CAM reference implementation and is in the same range. But several result values are above these values in the range of up to 50%, and some few up to 67%. The strong dispersion, compared to the ETSI *ITS-G5* CAM reference implementation reflects the missing of a DCC mechanism in our approach. Due to our optimized object tracking mechanism, the wireless channel is still not completely crowded, even with a very high vehicle density.

The average CBR is given in Figure 56 for all evaluation run configurations r.1 to r.18 and for the four vehicle densities of 250, 500, 1000 and 1500 vehicles. Since the confidence intervals are very small, these are poorly visible. Therefore we provide the respective evaluation result data in Table 46, in Section A.3.3. The figure compares the CBR values for our approach with CAM and CPM together, i. e., r.1 to r.18, with the ETSI *ITS-G5* CAM reference implementation. For vehicle densities of 250, 500 and 1000 vehicles in the scenario, our approach shows very homogenous CBR values over all run configurations. For the two scenarios with the lower vehicle density, i. e., 250 and 500 vehicles, the CBR values of our approach are almost half compared to the ETSI *ITS-G5* CAM reference implementation. For the scenario of 1000 vehicles, the CBR values are all comparable around 20%. In case of the highest vehicle density, i. e., the scenario of 1500 vehicles, the CBR values of our approach exceed the result values of the ETSI *ITS-G5* CAM reference implementation. For r.1 to r.9, i. e., with a minimum CAM sending rate of 1 Hz, the average CBR is in the range of 27%. In case of r.10 to r.18, i. e., with a minimum CAM sending rate of 0.5 Hz, the average CBR is in the range of 30%. This growth is due to the much larger size of CPMs compared to CAMs. For r.10 to r.18, the amount of CAMs decreases and thus, the amount of CPMs increases. The higher ratio of larger CPMs causes a higher CBR.

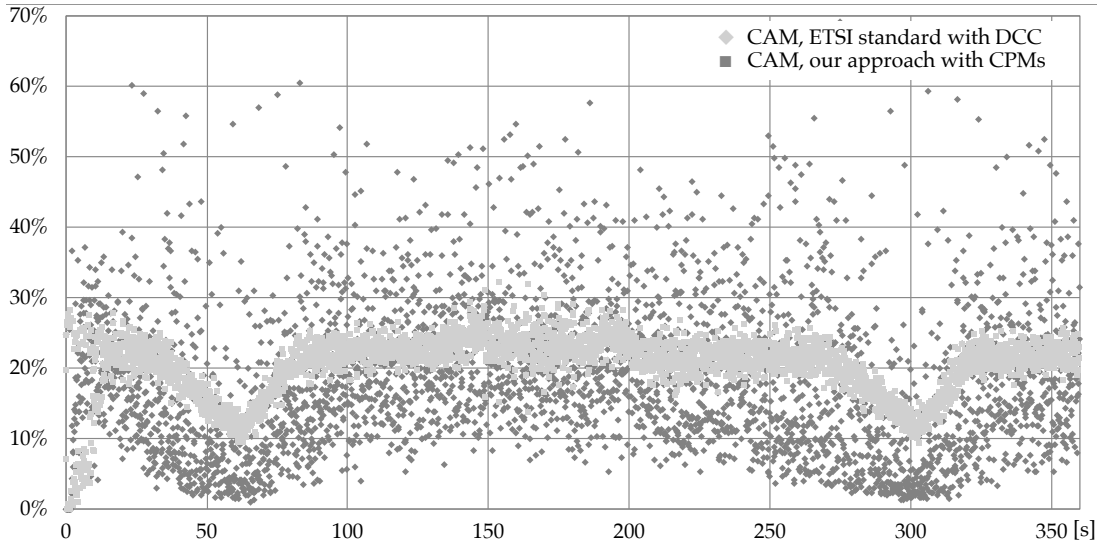

Figure 54: Example illustration of Channel Busy Ratio (CBR) with the use of CPMs over time for one example vehicle. The figure gives results for a scenario with 1000 vehicles. For our approach, the example shows run configuration r.10, i. e., a minimum CAM sending rate of 0.5 Hz, a tolerated prediction error of 0.2m and a threshold to switch the kinematic model of $0.3\frac{\mathrm{m}}{\mathrm{s}^2}$.

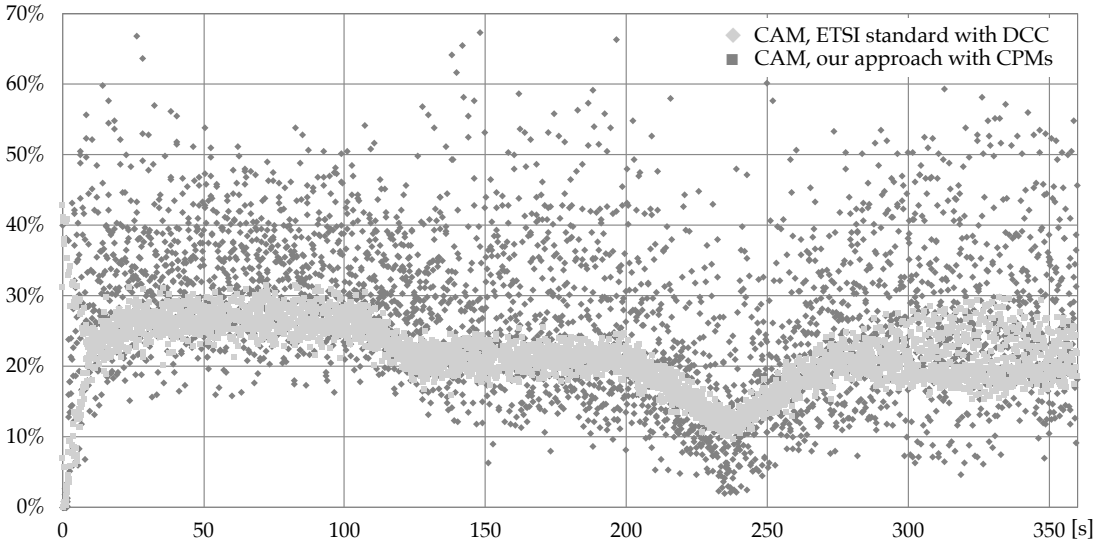

Figure 55: Example illustration of Channel Busy Ratio (CBR) with the use of CPMs over time for one example vehicle. The figure gives results for a scenario with 1500 vehicles. For our approach, the example shows run configuration r.10, i. e., a minimum CAM sending rate of 0.5 Hz, a tolerated prediction error of 0.2m and a threshold to switch the kinematic model of $0.3\frac{m}{s^2}$.

Analysis of the local knowledge: An indicator that describes the overall vehicular perception, with regard to moving objects, is the average amount of known other vehicles in the local vehicle database, that we compare for our approach with and without the use of CPMs. Also here, the confidence intervals are very small and poorly visible. Therefore, we provide the respective evaluation result data in Tables

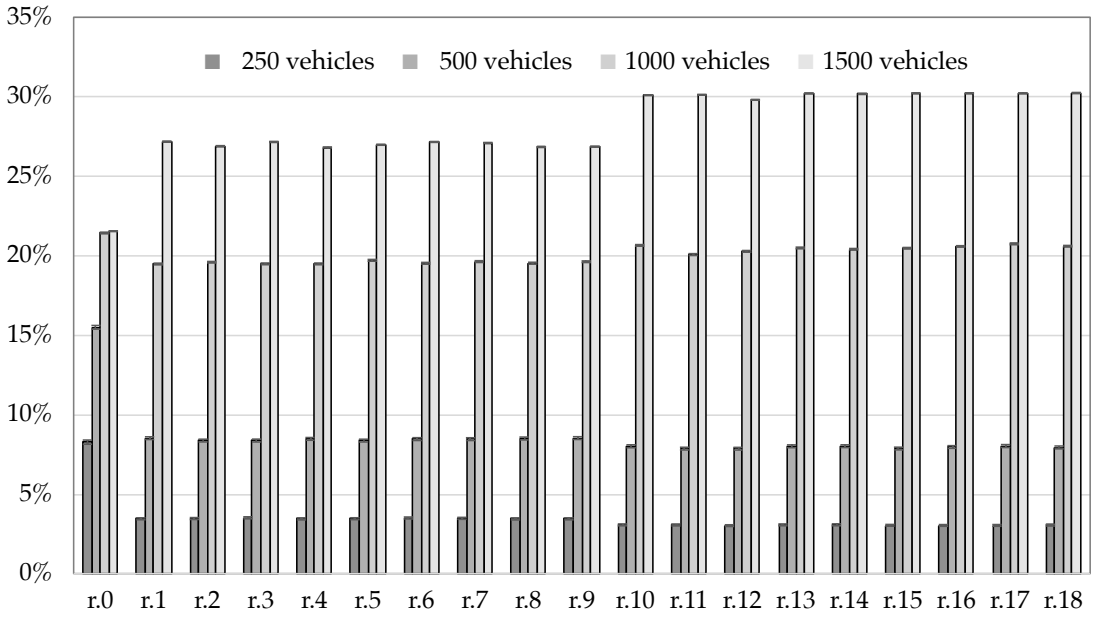

Figure 56: Average Channel Busy Ratio (CBR) with the use of CPMs. The figure gives results for all different evaluation run configurations r.1 to r.18, as described in Section 7.3.1.3 and listed in Table 14. Each, for the scenario with 250, 500, 1000 and 1500 vehicles.

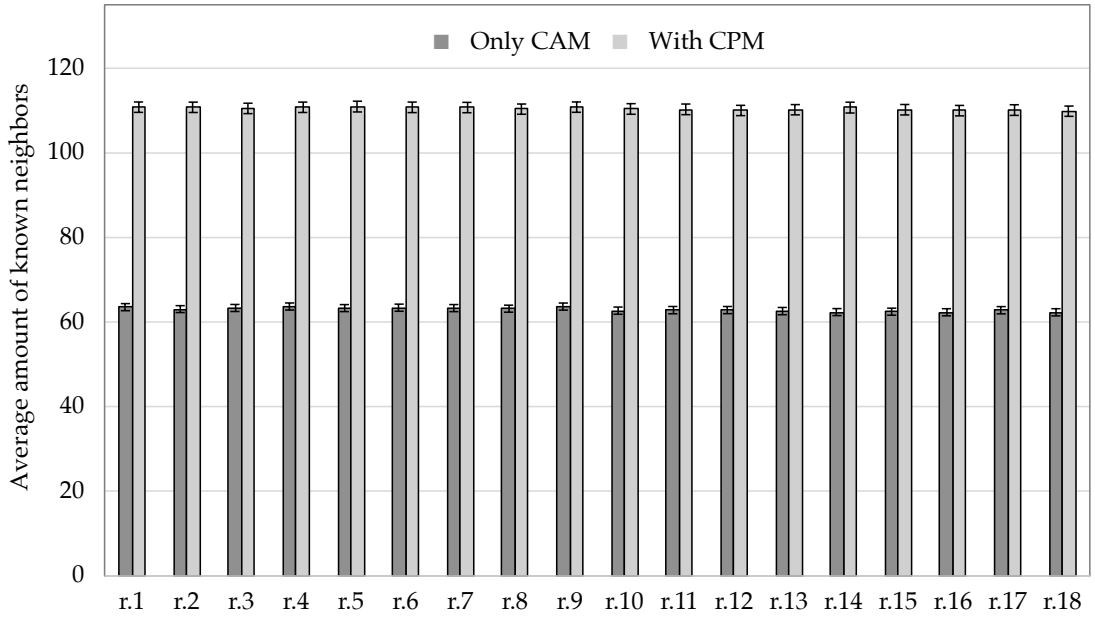

Figure 57: Average amount of known other vehicles in the local vehicle database, with and without the use of CPMs. The figure gives results for a scenario with 250 vehicles for all different evaluation run configurations r.1 to r.18, as described in Section 7.3.1.3 and listed in Table 14.

47, 48, 49 and 50 in Section A.3.3. The results for the lowest vehicle density, i. e., the scenario of 250 vehicles, are given in Figure 57. First of all it can be seen, that the amount of known neighbor vehicles is very homogenous over all different evaluation run configurations r.1 to r.18. Without the use of CPMs, the average amount of

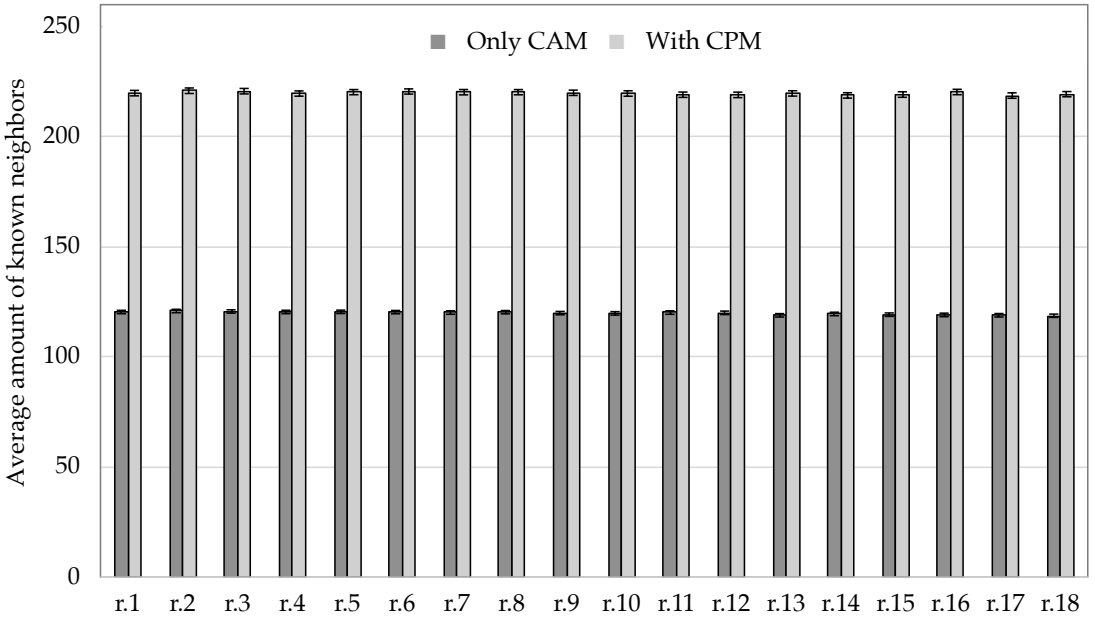

Figure 58: Average amount of known other vehicles in the local vehicle database, with and without the use of CPMs. The figure gives results for a scenario with 500 vehicles for all different evaluation run configurations r.1 to r.18, as described in Section 7.3.1.3 and listed in Table 14.

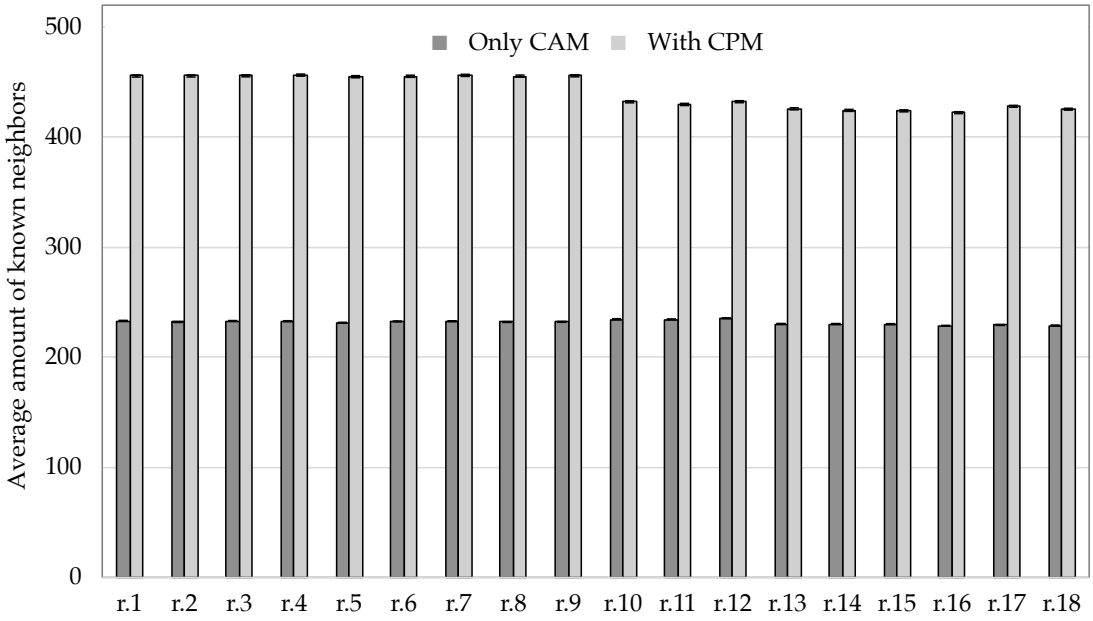

Figure 59: Average amount of known other vehicles in the local vehicle database, with and without the use of CPMs. The figure gives results for a scenario with 1000 vehicles for all different evaluation run configurations r.1 to r.18, as described in Section 7.3.1.3 and listed in Table 14.

known other vehicles in the local vehicle database is about 63. In case of using CPMs, we are able to almost double this value to about 110 vehicles. A similar result can be seen for the next higher vehicle density, i. e., the scenario of 500 vehicles, as depicted in Figure 58. The amount of known neighbor vehicles is very homogenous, with nearly same values over all evaluation run configurations r.1 to r.18. Without the use of CPMs, the average amount of known other vehicles in the local vehicle database is about 120. In case of using CPMs, we are able to almost double this value to about 220 vehicles. For the vehicle density of 1000 vehicles in the scenario, the result slightly differs, as depicted in Figure 59. Without the use of CPMs, the average amount of known other vehicles in the local vehicle database is still homogenous over all evaluation run configurations, with values of about 230 vehicles. In case of using CPMs, we are able to almost double this value to about 455 vehicles. But in case of using CPMs, we see a drop between r.1 to r.9 and r.10 to r.18 to an average amount of known other vehicles in the local vehicle database of about 425 vehicles. A reason for this effect is the maximum valid time of CAM information. In case of the lower CAM sending rate in r.10 to r.18, received vehicle information can be older and thus gets deleted earlier. Similar results can be seen for the highest vehicle density of 1500 vehicles in the scenario, as depicted in Figure 60. In this high vehicle density, the previously mentioned effect is almost bigger. This is due to a reduced average CAM sending rate, caused by a higher CBR and slightly reduced mobility, because of congested road segments. This effect is already visible in case of using only CAMs without CPMs. In this case, the drop in the average amount of known other vehicles in the local vehicle database is from about 330 vehicles for r.1 to r.9 to about 325 vehicles for r.10 to r.18. In case of using CPMs, we are able to almost double this value to about 655 vehicles for r.1 to r.9. For r.10 to r.18 we see a drop to an average amount of known other vehicles in the local vehicle database of about 570 vehicles.

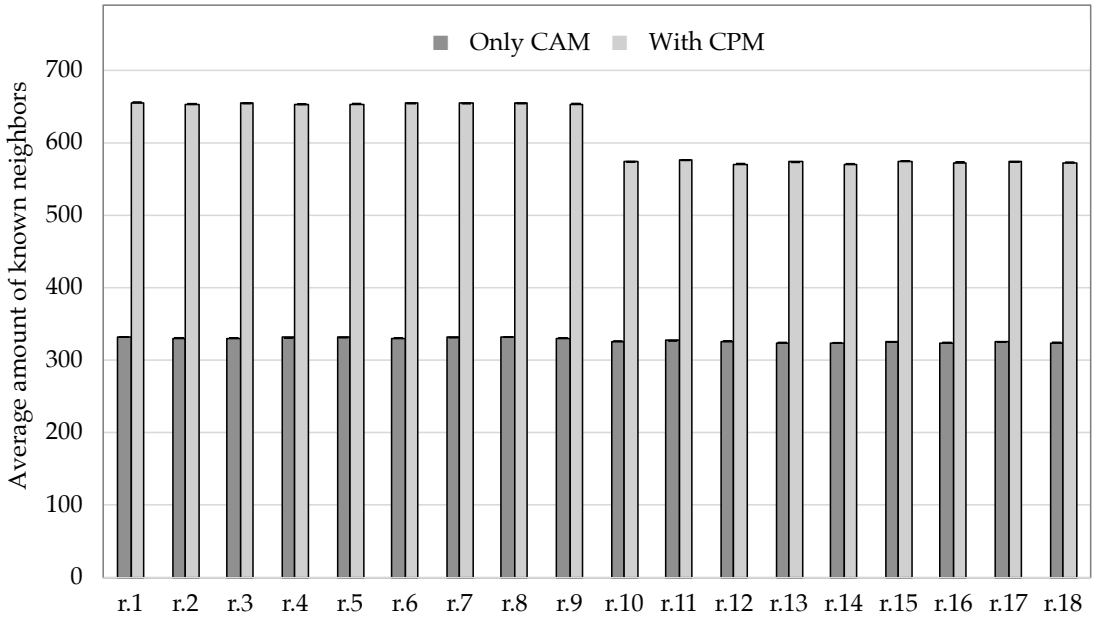

Figure 60: Average amount of known other vehicles in the local vehicle database, with and without the use of CPMs. The figure gives results for a scenario with 1500 vehicles for all different evaluation run configurations r.1 to r.18, as described in Section 7.3.1.3 and listed in Table 14.

Analysis of the packet error rate: Finally, we consider the PER in case of using CPMs. The evaluation results of the average PER of our approach with the use of CPMs for all four vehicle densities, are depicted in Figure 61. We provide the respective evalu-

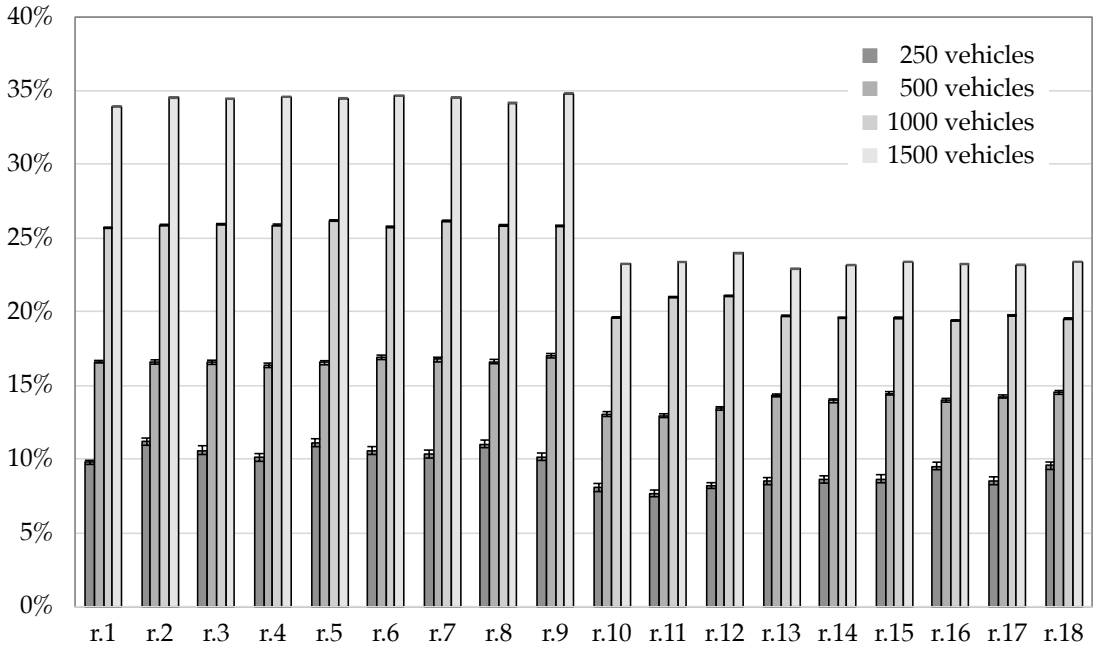

Figure 61: Average Packet Error Rate (PER) of our approach with the use of CPMs. The figure gives the results for all different evaluation run configurations r.1 to r.18, as described in Section 7.3.1.3 and listed in Table 14. Each, for the scenario with 250, 500, 1000 and 1500 vehicles.

ation result data in Table 51 in Section A.3.3. The results of the average PER of our approach without the use of CPMs for all four vehicle densities, are depicted in Figure 42 and have been previously discussed. For the scenario of 250 vehicles, the PER is in the range of about 10%, with 1% to 2% lower values for r.10 to r.18. We can see a linear increase of the PER with an increase of the vehicle density. For the scenario of 1500 vehicles, we can see an average PER of about 34% for run configurations r.1 to r.9, i. e., a minimum CAM sending rate of 1 Hz. Without the use of CPMs, as given in Figure 42, we can see a higher average PER of about 39%. This drop is due to the much larger size of CPMs compared to CAMs. The longer continuous channel occupancy, caused by the larger CPMs, reduces collisions caused by simultaneous transmission starts. For run configurations r.10 to r.18, we can see an average PER of about 23% for the scenario of 1500 vehicles. Compared to only using CAMs, this is an increase of about 3%, caused by the higher total number of messages in case of r.10 to r.18. However, the PER is strongly affected by the trigger check frequency, as we show in the following Section 7.3.2.3. The relatively simultaneous start of all simulated vehicles and the congruence of the trigger check interval with minimum CAM sending rate, leads to a not representative high PER.

7.3.2.3 *Influence of Trigger Check Frequency*

The used ETSI *ITS-G5* CAM reference implementation uses a trigger check interval of 10ms, despite the minimum CAM sending interval is 100ms. In our approach, as presented in the previous Sections, we used a trigger check interval of 100ms, due to the high computation time in the simulations. To analyze the effect of an increased trigger check frequency from 10 Hz to 100 Hz, we consider the number of generated CAMs, the CBR and the PER in the scenario of 500 vehicles in the following. Therefore, we have executed our evaluations again for the scenario with 500 vehicles, this time with a trigger check frequency of 100 Hz, i. e., a trigger check interval of 10ms. First of all, we start with a comparison of the average number of generated CAM messages, with a trigger check interval of 100ms and 10ms. As a reference, the standardized ETSI *ITS-G5* CAM implementation, i. e., r.0, needs about 2373 CA messages in this scenario, as given above in Figure 36 and in Table 29 in Section A.3.3. Figure 62 shows the evaluation results of the average number of generated CAM messages for a scenario with 500 vehicles, with a trigger check interval of 100ms and 10ms. We provide the respective evaluation result data in Table 52 in Section A.3.3. The figure shows an increase of the average number of generated CAM messages of 7% to 20%, for run configurations r.10 to r.12 even up to 39%. With the higher trigger check frequency, the system is able to react more dynamically on changes in the vehicle dynamics. As a consequence thereof, the points in time individual vehicles trying to send new CAMs are more equally distributed, which reduces transmission collision risk. These effects can be seen in the Channel Busy Ratio (CBR) and Packet Error Rate (PER), as considered in the following.

Analysis of the channel busy rate: The next considered performance parameter is the CBR for a trigger check interval of 10ms. As reference, the standardized ETSI *ITS-G5* CAM implementation, i. e., run configuration r.0, has in this scenario a CBR of about 15%, as given above in Figure 41 and in Table 33 in Section A.3.3. This configuration also uses a trigger check interval of 10ms. The other results in Figure 41, i. e., run configuration r.1 to r.18, show the respective results for a trigger check

interval of 100ms. The respective results with an increased trigger check interval of 10ms, are given in Figure 63 for the scenario with 500 vehicles. For broadcasting CAMs with our approach, we can see a slight increase of the CBR from about 2.2% to $2.2 - 2.6\%$ for run configurations r.1 to r.9, and from about 1.2% to $1.3 - 1.6\%$ for run configurations r.10 to r.18. This increase corresponds to the increase in the number of sent CA messages, as previously shown. Figure 63 also shows the CBR regarding our CAM forwarding mechanism, i. e., broadcasting CPMs, with a trigger check interval of 10ms. The respective results for a trigger check interval of 100ms have been previously shown in Figure 56. In case of the higher trigger check interval, we can see almost similar CBR values around 8%. For run configurations r.1 to r.6, we can even see lower values of about 7%, which is a decrease of about 1.25% to 1.45% compared to sending CPMs with a higher trigger check interval of 100ms. For run configurations r.7 to r.9, we can even see about 0.45% higher values and for r10 to r.18, the results are almost the same compared to sending CPMs with a higher trigger check interval of 100ms. Overall, we can see that the CBR is not significantly affected by the trigger check frequency.

Analysis of the packet error rate: Slightly different results can be seen for the PER, as considered in the following. Here, we can see a considerably strong decrease of the PER, compared to the higher trigger check interval of 100ms. As a reference, the standardized ETSI *ITS-G5* CAM implementation, i. e., run configuration r.0, has a CBR of about 9.4% in this scenario, as given above in Figure 42 and in Table 36 in Section A.3.3. This configuration also uses a trigger check interval of 10ms. The other results in Figure 42, i. e., run configuration r.1 to r.18, show the respective results for a trigger check interval of 100ms. The respective results with an increased trigger check interval of 10ms, are given in Figure 64 for the scenario with 500 vehicles. In case of only broadcasting CAMs, we can see for r.1 to r.6 a strong decrease of the

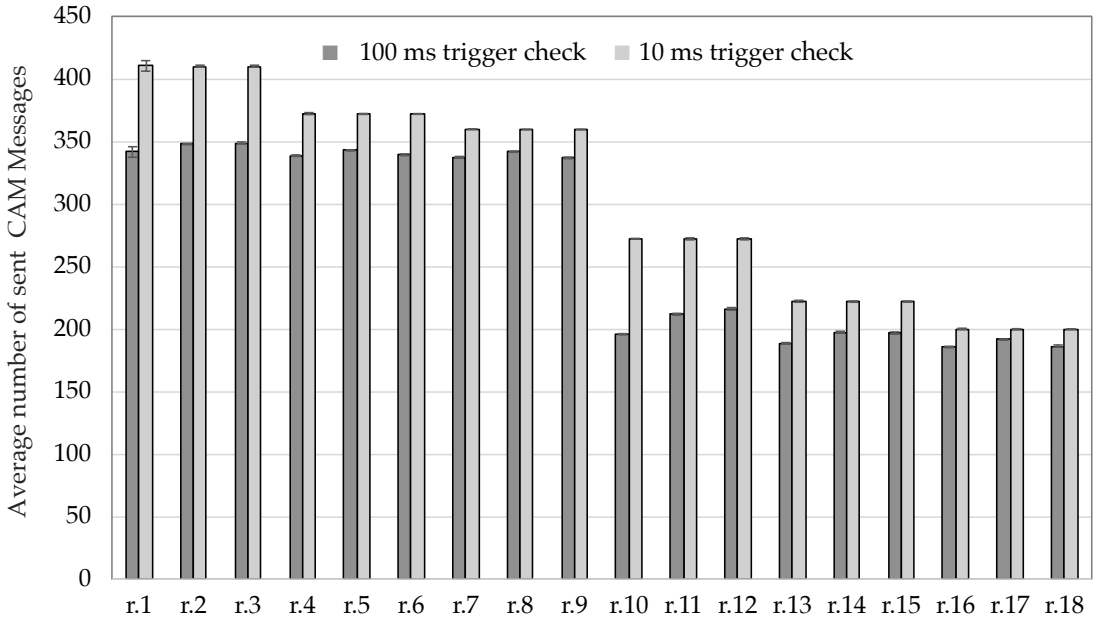

Figure 62: Comparison of the average number of generated CAM messages with a trigger check interval of 100ms and 10ms. The figure gives the results for a scenario with 500 vehicles for all different evaluation run configurations r.1 to r.18, as described in Section 7.3.1.3 and listed in Table 14.

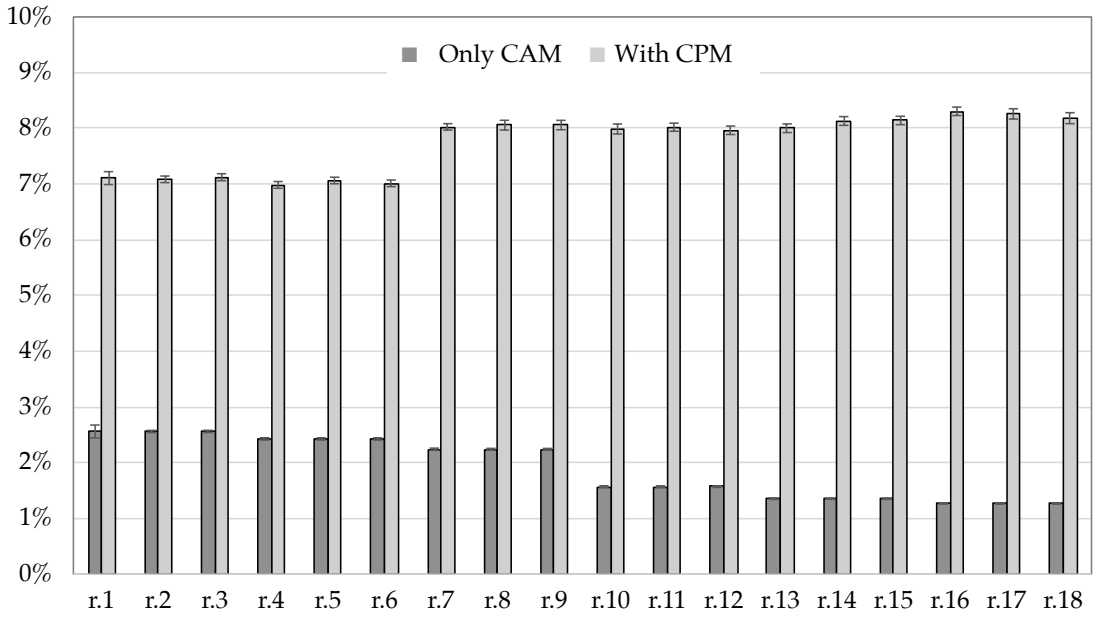

Figure 63: Comparison of the average CBR for broadcasting only CAMs and CAMs together with CPMs, with a trigger check interval of 10ms. The figure gives the results for a scenario with 500 vehicles for all different evaluation run configurations r.1 to r.18, as described in Section 7.3.1.3 and listed in Table 14.

PER from about 15% to about 3% and for r.10 to r.18 a decrease from about $8-10\%$ to about 3.5%, compared to a trigger check interval of 100ms. Exceptions are the results of r.7 to r.9, which show a slight increase of the PER of about 1.5%, from values of about 15% to about 16%. A similar decrease of the PER can be seen in case of using additionally CPMs. For the 10ms trigger check interval, we can see for r.1 to r.6 a strong decrease of the PER from about 17% to about 6%, compared to a trigger check interval of 100ms. In case of r.10 to r.12, the results remain relatively the same with values of about 13%. For r.7 to r.9 and for r.13 to r.18, we can see a slight increase in PER of about 1%. Run configurations r.7 to r.9 have the highest tolerated prediction error of 1m. The results show that a high tolerated prediction error may cause simultaneous transmissions. This might be caused by a specific road geometry or moving pattern, which causes a simultaneous prediction error trigger. Overall, the results show the strong influence of the trigger check rate with regard to the PER. However, this might be caused by the simulation tools and setup.

7.3.2.4 *Data Accuracy and Granularity*

In real world, sensor values are commonly not absolutely correct, but containing some errors. These errors are caused by interference and sensor noise. As an example, positioning via GNSS has a certain inaccuracy. This inaccuracy depends on system related components, e. g., antenna quality, and dynamic atmospheric effects. The same holds true for all measurements, also for heading, acceleration and velocity. Typically this inaccuracy can be modeled as a variance with a certain probability distribution. For instance *GPS* noise is Gaussian distributed plus a certain shift in consecutive measurements, caused by satellite, atmospheric and receiver effects [160, 161]. By the use of additional ground stations, a commercial correction signal, named

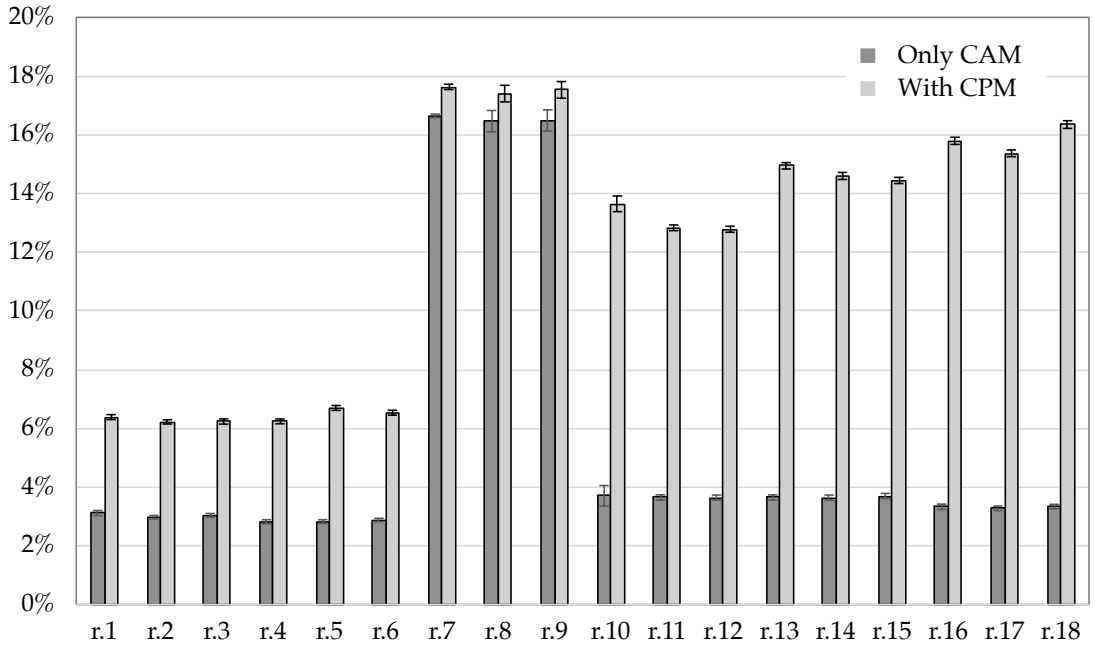

Figure 64: Comparison of the average PER for broadcasting only CAMs and CAMs together with CPMs, with a trigger check interval of 10ms. The figure gives the results for a scenario with 500 vehicles for all different evaluation run configurations r.1 to r.18, as described in Section 7.3.1.3 and listed in Table 14.

Differential Global Positioning System (DGPS), is generated. This can be used to increase *GPS* accuracy in the range of some centimeters [162]. Additionally processing delays would occur in a real world system and map material will be mostly not free from errors.

Each error source causes the need of an own error model, which leads to high complexity. On the other hand, the magnitude of error can be significantly reduced by high quality sensors, sensor fusion and the use of commercial correction services like DGPS. Thus, we have not considered theses measurement errors in our evaluations. In our considerations the focus is on communication and not on measurement errors. But in exchange we have set the parameter for the tolerated prediction error ε to relatively low values of 0.2m, 0.5m and 1m.

Moreover, in our simulations the map material is also provider for position information. This leads to a 100% accuracy in positioning with 100% congruency. In a real world setup this is not realistic. To show the behavior of our system in comparison to real world conditions, we have used a recorded real world vehicular DGPS trace. The trace was provided by Daimler AG, Stuttgart. A test vehicle, equipped with DGPS and recording hardware, performed a test drive of 23.3km in total, as depicted in Figure 65. The vehicle started at exit 21 *Sindelfingen-Ost*, continued on *A81* in direction *Singen*, used the exit 24 at *Böblingen-Hulb* and continued on *A81* in direction *Stuttgart*, used again exit 21 *Sindelfingen-Ost* continued again on *A81* in direction *Singen*, used again exit 24 at *Böblingen-Hulb*, continued on *Böblinger Straße* and then turned left onto *Gottlieb-Daimler Straße*. The recorded trace is composed of 17.6km motorway, 4.7km feeder road and freeway interchange as well as 1km urban road. The total recording time is 1120 seconds, i. e., about 18.7 minutes, with a position value every 0.1 seconds. Since it is a trace of a single vehicle, we can not analyze channel load

effects caused by neighboring vehicles. But we compare the amount of sent CAMs according to the ETSI *ITS-G5* standard CA service and our approach. Therefore, we use the described trace as input for the CAM sending mechanism. In Figure 66 we give the results of sent CAM messages, as number of generated messages over time, for r.0, the CAM sending mechanism according to standardized ETSI *ITS-G5* implementation, and r.1 and r.10 for our approach as comparison. For position prediction in our approach we still stick to the simple map source as previously described. We have selected r.0 and r.10, with $T_{CAM,min}$ of 0.5 Hz and 1 Hz as an example configuration of our approach, since the minimum CAM sending rate has the strongest impact, according to the amount of sent messages. The figure shows for r.0 a total amount of 5029 sent CA messages. This corresponds to an average sending rate of approximately 5 Hz and correlates with our previously presented results of our simulations. For our approach in configuration r.1 we get 1760 sent CA messages and in r.0 we get 1536 sent CA messages. This shows a reduction of sent CA messages of up to 70%. These results show the transferability of our approach towards a real world scenario.

7.3.2.5 *Discussion*

The primary intention of our communication based object tracking mechanism is to free-up bandwidth, which is used in turn to extend the tracking range by information forwarding. This enables a significant increase of the vehicular perception range, with respect to moving objects. In our approach, we combine two kinematic models with a map based heading adjustment to realize trajectory prediction. Based on the this prediction, we control the transmission rate of position beacons, i. e., CAMs. The sender observes the accuracy of an ego position estimator, that is based on the prediction model, and the measured position. Input of the position estimator is the last broadcasted CAM. The used prediction module on the sender side is the same as on the receiver side. Thus, the sender estimates the difference between the measured ego position and the position, a neighbor vehicle would estimate. Next, a new CAM is only broadcasted if the calculated difference violates a given threshold or a timer exceeds. Thus, our aim is not to prevent channel congestion with a high PER, but to proactively reduce the overall channel load to a minimum. Related approaches in literature are typically related to safety relevant use cases and tend to achieve a

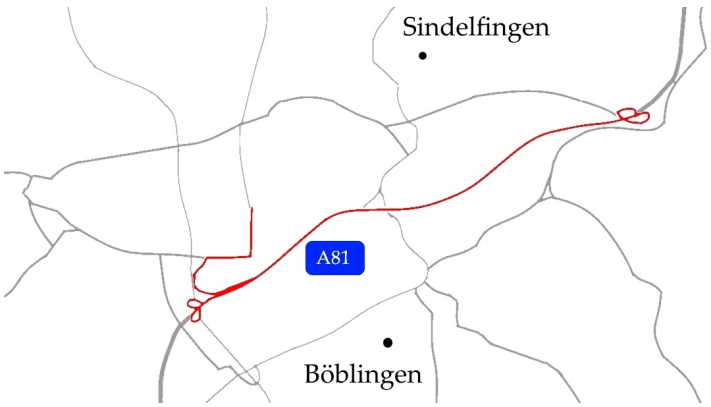

Figure 65: Visualization of the route of the used DGPS trace.

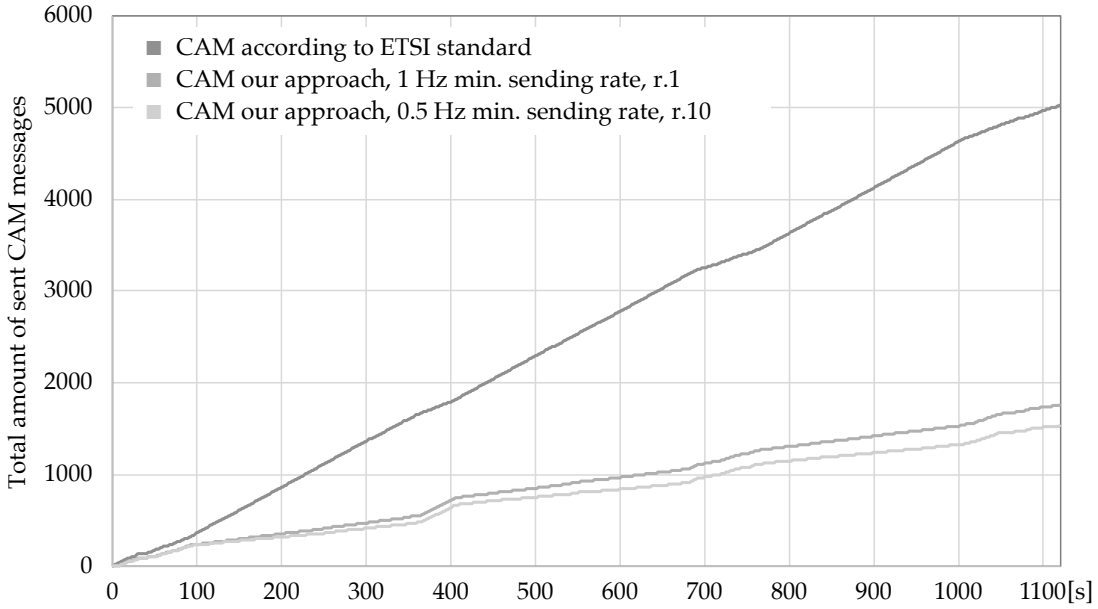

Figure 66: Illustration of the number of generated CAM messages over time for one example vehicle, based on the used DGPS trace. Comparison of standardized ETSI *ITS-G5* CAM implementation to our approach. For our approach we have selected run configuration r.1 and r.10 (cf. Table 14), which have a minimum CAM sending rate of 0.5 Hz and 1 Hz, each with a tolerated prediction error of 0.2m and a threshold, to switch kinematic model, of $0.3\frac{m}{s^2}$.

high CAM, or BSM, transmission rate [79, 94, 80]. The use of trajectory prediction, to control the transmission rate, is also used in other approaches, but only based on a single kinematic model and to increase a basic transmission rate, in case of high vehicle dynamics [93, 94]. Other approaches additionally decrease the transmit power, to avoid channel congestion [56]. However, this would be contrary in our approach of increasing the perception range.

Our used simulation scenario is exactly the same as in [94, 80], which allows us to compare some evaluation metrics. Our primary aim is to free-up bandwidth, that can be typically measured as CBR, also named channel busy percentage in [94, 79]. In case of the standardized ETSI *ITS-G5* CAM implementation, we can see an average CBR of about 21%, which also corresponds to the results shown in [94, 80]. With our approach, we are able to reduce the CBR in all cases to values below 4%, which results in about 80% less channel load.

Another metric that reflects the average channel load is the average CAM sending rate, related to the Inter Transmit Time (ITT) in [94]. In case of the standardized ETSI *ITS-G5* CAM implementation, we can see values of up to about 7 Hz. It is decreasing to about 3 Hz, in case of a higher channel load, which reflects the attempt to achieve a high transmission rate. Other approaches, that also use trajectory prediction, like *InterVehicle Transmit Rate Control (IVTRC)* [93] or *InterVehicle Transmit Rate and Power Control (IVTR-PC)* [56], show in this scenario an average CAM sending rate of about 3.6 Hz for all vehicle densities [94]. Approaches, that focus more to the channel load itself, like *Linear Integrated Message Rate Control (LIMERIC)* [80] or Error Model Based Adaptive Rate Control (EMBARC)[94], show in our considered scenario an average CAM sending rate between 10 Hz and 2 Hz, depending on the channel

load, respectively vehicle density. In case of our approach, we are able to reduce the average CAM sending rate down to about 0.5 Hz, independent of the vehicle density without losing tracking accuracy. To get an impression of real world behavior of our CAM mechanism, we used a real world DGPS trace. The respective results still show a very promising low average CAM transmission rate of about 1.37 Hz.

To obtain our primary aim of an extended perception range, we introduced the Cooperative Perception Message (CPM), that is broadcasted by each vehicle to forward local perception knowledge. Thus, the freed-up bandwidth is used to forward tracking information. The basic principle is to use a local position estimator for each known vehicle, that is overridden each time recent information of the respective vehicle is received. If no position update is received for a certain time, the respective vehicle is removed from the local perception knowledge. With this approach, we achieve an average completeness of the local perception knowledge of more than 90%, even in the high traffic density scenario of 1500 vehicles.

As a result, we can see an increased CBR and PER. But with respect to the PER, we have identified an insufficient trigger check rate as major cause. This resulted in relative synchronicity of the simulated vehicles and as a consequence, a high PER. Hence, we have repeated our experiments with a higher trigger check interval of 10ms, according to the ETSI implementation. The results show a significant decrease of the PER. Even in case of broadcasting CPMs in addition to CAMs, we achieve a significantly lower PER, compared to the standardized ETSI *ITS-G5* CAM broadcast approach. Whereas the ETSI implementation has an average PER of about 10% (c.f. Figure 42) in the scenario of 500 vehicles, our CAM approach shows an average PER of only about 3%. In case of broadcasting CPMs in addition, we can see a doubled PER of about 6% that is still significantly lower in comparison to the ETSI approach. According to the better PER results, compared to the standardized ETSI *ITS-G5* implementation, we assume our approach also as robust enough for safety relevant use cases.

To verify our results under real world conditions, an extensive field test would be the next step. However, this would require extensive resources that are not available within this work. An improvement to ensure global fairness, with respect to wireless channel usage, could be to add CBR information to the CPMs and the use of a respective local channel congestion management, similar as shown in [94]. But due to the overall significantly reduced channel load, this should be not necessary.

As shown in our evaluation, with our approach we are able to significantly reduce the amount of CA messages down to rates of about 0.5 Hz. Due to the significantly reduced amount of messages, a respective object tracking might also be possible via cellular networks. According to [163], such low CAM rates would allow a single LTE base station to support about 1200 vehicles in an infrastructure based approach. A possible scenario, applying to the novel paradigm of edge computing, would be an edge server (also known as cloudlet), directly connected to the cell tower (*eNB*), that provides long-range perception based services [65, 164, 165].

8

CONCLUSIONS AND OUTLOOK

Today's vehicular Advanced Driver Assistance Systems (ADASs) are limited by the physical sensing range of built-in sensors. Within this thesis, we presented our concept to extend this perception range, to provide the means to enhance existing ADASs and even enable completely new ADAS concepts, including more predictive systems. Within this final chapter, we first summarize the main contents and contributions of this thesis in section 8.1, followed by a discussion of potential future work in Section 8.2.

8.1 SUMMARY AND CONCLUSIONS

Advanced Driver Assistance Systems (ADAS) become increasingly important, due to growing vehicular traffic, that is getting more dense and complex, with more accidents and a higher environmental impact [2, 3, 4]. In this context, we have identified the bounded perception range of built-in sensors, as a limiting factor for the capabilities of ADASs. An extension of this perception range is required, in order to enhance existing or even enable completely new ADAS concepts. This will enable more predictive systems, to increase traffic efficiency and driving comfort, economy and safety. Thus, our overall aim within this thesis is the extension of the vehicular perception range, to provide the information basis for future ADASs. The respective research questions that we have addressed in this thesis are: *How to collect perception information efficiently?*, *Which information is relevant?* and *How to efficiently distribute respective information?*

In a first step, we started with a comprehensive analysis of connected vehicle applications and use cases from European Telecommunications Standards Institute (ETSI) documents and relevant research publications. Based on this, we have restructured the described use cases into a harmonized categorization scheme. This scheme consists of three zones of information demands, namely *safety zone*, *awareness zone* and *information zone*. In addition, we have discussed aspects of the communication range of vehicular ad-hoc networks and derived the feasible communication technologies for the three zones of information demands. Whereas the *safety zone* is out of scope of this thesis, we have introduced appropriate communication solutions for the *awareness zone* and the *information zone*.

With regard to the *information zone*, we propose a logically central approach, that consists of a probabilistic information gathering approach, as introduced in Chapter 5, and an information distribution approach as cloud-based eHorizon, as introduced in Chapter 6. With regard to the *awareness zone*, we identified the need of long-range information about moving objects, i. e., other vehicles. The proposed solution uses information forwarding, based on vehicular ad-hoc communication, as introduced in Chapter 7. As a prerequisite, we introduced an optimized communication scheme for tracking neighboring vehicles, since state-of-the-art communication approaches use up the complete wireless channel in, case of dense traffic.

Our probabilistic information gathering approach intents to minimize the cellular data traffic and thus, minimize transmission costs. Concurrently, a certain maximum desired detection latency is guaranteed, since the data quality depends on the latency it takes to transmit a certain information to the backend, with a defined redundancy. We have named our proposed approach *ProbSense.KOM*, and have also introduced the extension *Hybrid-ProbSense.KOM*, that is based on hybrid communication, i. e., the use of cellular networks and vehicular ad-hoc communication. In contrast to classic data collection approaches, that use local pre-processing or clustering with aggregation to reduce the data traffic, our approach makes use of an incomplete transmission model. The potentially high amount of redundancy in the transmitted data is reduced by transmission probabilities, that are adjusted by a backend. The appropriate transmission probability is calculated based on the received data rate, to minimize the total amount of data traffic and concurrently stick to a certain maximum desired detection latency. This adjustment of the transmission probabilities is calculated individually for each event type and separately for each geo cell. A certain maximum desired detection latency is guaranteed, as long as a sufficiently high vehicular traffic density allows enough event detections. Periodically adjustments enable to cope with a continuously varying traffic density over time. Thus, the backend serves as an always up-to-date information source, which is the basis to provide a cloud-based eHorizon to extend the vehicular perception. We have evaluated our approach with the open available *TAPAS Cologne* and *Luxembourg LUST* scenarios in the SUMO traffic simulator. The results have shown a possible reduction in the total cellular data traffic of about 55%, compared to an opportunistic transmission model. In case of the availability of additional vehicular ad-hoc communication, our hybrid communication model *Hybrid-ProbSense.KOM* even shows a possible reduction of up to 82%. Even a penetration rate with vehicular ad-hoc communication of about 25% shows significant reductions and a penetration rate of about 50% is enough to achieve roughly the maximum possible reduction of cellularly transmitted data.

To provide this logically central up-to-date information towards the ADASs in the vehicles, we propose our cloud-based eHorizon provider, that we have also provided under an open source license to the community. The system is able to segmentally transmit an eHorizon and makes use of an asynchronous robust communication approach. The concept allows to transmit only data that is relevant for the current trip and makes constant map updates unnecessary. Our concept also includes a heuristic Most Probable Path (MPP) determination, which allows to provide an eHorizon without any knowledge about the requesting vehicle. However, knowledge about the MPP from historic data or from the navigation system allows to decrease the necessary amount of transmitted data.

Regarding the *awareness zone*, more dynamic information is required, i. e., long-range information about other vehicles. Due to the high amount of required data, we propose direct inter-vehicle communication, with an according information forwarding, based on vehicular ad-hoc communication. However, in case of dense traffic, state-of-the-art communication approaches for tracking neighboring vehicles use up the complete wireless channel, even in combination with channel load control mechanisms. Thus, a prerequisite to enable the forwarding of dynamic vehicle information, is an overall optimized approach to communicate tracking related vehicle information. In this context, we have introduced a communication based object tracking mechanism, that can significantly reduce the channel load. The sender vehicle

estimates the entropy in a new position message. Our general concept makes use of trajectory prediction, that we have realized with the combination of two kinematic models, with a map based heading adjustment. The transmission of new position beacons is controlled, based on an estimated error in the trajectory prediction. All vehicles run a position estimator for each known neighbor vehicle, based on the last received position beacon. Moreover, each vehicle makes use of an ego-position estimator, that reflects the position, a neighbor would estimate. This position estimator is exactly the same as used in the neighbor vehicles and uses only the last transmitted position beacon as input. Hence, a vehicle runs locally the same position estimator for its own position, as the neighbors. By a comparison of the measured own position and the output of the ego-position estimator, a vehicle can estimate the error of the own position in the neighboring vehicles' position estimation. A new position beacon is only broadcasted, in case the calculated difference violates a given threshold or a timer exceeds. State-of-the-art solutions prevent channel congestion with a high Packet Error Rate (PER), but intend a constantly high channel load. In contrast to this, we proactively reduce the overall channel load. We have done extensive simulations in a network simulator. With our approach, we are able to reduce the Channel Busy Ratio (CBR) in all settings to values below 4%, which results in about 80% less channel load. A real-world comparison, based on a differential *GPS* trace, has shown promising results with an average position beacon transmission rate of about 1.37 Hz. Additionally, our approach makes use of the freed-up bandwidth to forward local perception information about neighbor vehicles. Each vehicle uses a position estimator for each known vehicle, also for those known from forwarded information, that is overridden each time recent information of the respective vehicle is received. If no position update is received for a certain time, the respective vehicle is removed from the local perception knowledge. In the simulated scenario, our approach has shown an average completeness of the local perception knowledge of more than 90% in 1000 m distance, even in case of a very high vehicle density. Moreover, our simulations have shown a lower Packet Error Rate (PER) compared to the standardized ETSI *ITS-G5* implementation. Hence, we assume our approach as robust enough for safety relevant use cases.

In summary, we can conclude, that the combination of our introduced mechanisms provide the means for a holistic extension of the vehicular perception range. Each use case category has its own demands, that have been addressed with an appropriate solution.

8.2 OUTLOOK

As the initial step, to consider the demands of ADASs, we analyzed connected vehicle applications and use cases. Some of these use cases already deal with future needs of cooperative driving maneuvers and completely autonomous vehicles. However, until now, fully autonomous driving is not yet applied in real traffic. This results in a lack of experience, that will probably bring several new use cases and application demands in the future. This may result in a need of additional communication strategies.

With respect to our probabilistic information gathering approach, a further extension might be the consideration of fairness, in case of the hybrid communication

approach. Some vehicles might have unfair low cellular communication costs. We have not considered the fairness aspect, since our aim was to reduce the overall cellular data traffic.

To provide the gathered up-to-date information back to the vehicles, we propose a cloud-based eHorizon provider. We assume such a service to be logically central, but physically distributed. Comparable to Content Delivery Networks (CDNs), used for multimedia distribution, an appropriate mechanism distributes respective requests towards the best suited server instance, typically located in proximity to the requester. The design of such a highly scaleable backend concept was not scope of our work. However, we have introduced a cloud based service framework, to support mobility data based services in [17]. A future extension might be the integration of our eHorizon and probabilistic information gathering approaches. Moreover, also object tracking based cloud services might be possible, by making use of edge computing and in-network intelligence [164, 165]. In addition, cellular communication is bidirectional, in contrast to vehicular ad-hoc communication, with link layer error correction, which results in a more reliable connection with a lower Packet Error Rate (PER). Our introduced object tracking approach requires much less bandwidth, than traditional approaches, which could make the approach an enabler for tracking based services.

Finally, a required next step would be to verify our results under real-world conditions in an extensive field test. However, this would require extensive resources, that have not been available within this work.

8.3 ACKNOWLEDGEMENTS

We like to thank Niels Nötzel and Christoph Peusens for supporting us in the design and rendering of the illustrations used within this work. Moreover, we like to thank Gaurav Bansal, Bin Cheng, Ali Rostami, Katrin Sjoberg, John B. Kenney and Marco Gruteser for providing the SUMO simulation files to us, that we have used in Chapter 7. Finally, we like to thank Daimler AG, Stuttgart, for providing us with differential *GPS* trace records.

BIBLIOGRAPHY

[1] Y. Fang. "Connected Vehicles Make Transportation Faster, Safer, Smarter, and Greener!" In: *IEEE Transactions on Vehicular Technology (VT)* 64.12 (Dec. 2015), pp. 5409–5410. ISSN: 0018-9545.

[2] European Automobile Manufacturers' Association (ACEA). *World Production - 91.5 million motor vehicles were produced globally in 2015*. Accessed February 2017, http://www.acea.be/statistics/tag/category/world-production.

[3] Organisation Internationale des Constructeurs d'Automobiles (OICA). *Production Statistics*. Accessed February 2017, http://www.oica.net/category/production-statistics/.

[4] World Health Organization (WHO). *Global Status Report on Road Safety 2015*. ISBN: 978-92-4-156506-6.

[5] M. Möbus, M. Jacobs, M. Wagner, and S. Durekovic. "Digitale Karten als vorausschauende Sensoren für Fahrerassistenzsysteme." In: *GMM-Fachbericht-AmE 2011–Automotive meets Electronics* (Mar. 2011), pp. 111–115.

[6] C. Ress, A. Etemad, D. Kuck, and J. Requejo. "Electronic Horizon - Providing Digital Map Data for ADAS Applications." In: *Proceedings of the 2nd International Workshop on Intelligent Vehicle Control Systems (IVCS)*. 2008, pp. 40–49.

[7] TomTom International B.V. *TomTom Annual Report 2015*. Accessed February 2017, http://annualreport2015.tomtom.com/docs/TomTom2015/downloads/TomTom_Annual_Report_2015.pdf.

[8] W. Wachenfeld, H. Winner, J. C. Gerdes, B. Lenz, M. Maurer, S. Beiker, E. Frädrich, and T. Winkle. "Use Cases for Autonomous Driving." In: *Autonomous Driving: Technical, Legal and Social Aspects*. Springer, 2016, pp. 9–37. ISBN: 978-3-662-48847-8.

[9] S. Abdelhamid, H. S. Hassanein, and G. Takahara. "Vehicle as a resource (VaaR)." In: *IEEE Network* 29.1 (Jan. 2015), pp. 12–17. ISSN: 0890-8044.

[10] D. Jiang and L. Delgrossi. "IEEE 802.11p: Towards an International Standard for Wireless Access in Vehicular Environments." In: *Proceedings of the 2008 IEEE 67th Vehicular Technology Conference (VTC2008-Spring)*. May 2008, pp. 2036–2040.

[11] D. Burgstahler, M. Pelzer, A. Lotz, F. Knapp, H. Pu, T. Rückelt, and R. Steinmetz. "A Concept for a C2X-based Crossroad Assistant." In: *Proceedings of the 2nd IEEE PerCom Workshop on Smart Environments: Closing the Loop (SmartE 2015)*. IEEE. Mar. 2015. ISBN: 978-1-4799-8425-1.

[12] D. Burgstahler, F. Knapp, S. Zöller, T. Rückelt, and R. Steinmetz. "Where is That Car Parked? A Wireless Sensor Network-Based Approach to Detect Car Positions." In: *Proceedings of the 9th IEEE LCN International Workshop on Practical Issues in Building Sensor Network Applications (IEEE SenseApp 2014)*. IEEE. Sept. 2014, pp. 514–522. ISBN: 978-1-4799-3782-0.

[13] F. Dressler, H. Hartenstein, O. Altintas, and O. K. Tonguz. "Inter-vehicle communication: Quo vadis." In: *IEEE Communications Magazine* 52.6 (June 2014), pp. 170–177. ISSN: 0163-6804.

[14] X. Wu, S. Subramanian, R. Guha, R. G. White, J. Li, K. W. Lu, A. Bucceri, and T. Zhang. "Vehicular Communications Using DSRC: Challenges, Enhancements, and Evolution." In: *IEEE Journal on Selected Areas in Communications* 31.9 (Sept. 2013), pp. 399–408. ISSN: 0733-8716.

[15] K. Zheng, Q. Zheng, P. Chatzimisios, W. Xiang, and Y. Zhou. "Heterogeneous Vehicular Networking: A Survey on Architecture, Challenges, and Solutions." In: *IEEE Communications Surveys Tutorials* 17.4 (2015), pp. 2377–2396. ISSN: 1553-877X.

[16] C. Han, M. Dianati, R. Tafazolli, R. Kernchen, and X. Shen. "Analytical Study of the IEEE 802.11p MAC Sublayer in Vehicular Networks." In: *IEEE Transactions on Intelligent Transportation Systems* 13.2 (June 2012), pp. 873–886. ISSN: 1524-9050.

[17] D. Burgstahler, S. Schulte, S. Abels, K. Kipp, P. Hoenisch, S. Dustdar, and R. Steinmetz. "Informationssysteme für Verkehrsteilnehmer: Datenintegration, Cloud-Dienste und der Persönliche Mobilitätsassistent." In: *PIK - Praxis der Informationsverarbeitung und Kommunikation* 37.3 (Sept. 2014), pp. 243–250. ISSN: 0930-5157.

[18] *ETSI EN 302 665 V1.1.1 (2010-09) Intelligent Transport Systems (ITS); Communications Architecture*.

[19] H. Winner, S. Hakuli, and G. Wolf, eds. *Handbuch Fahrerassistenzsysteme*. Wiesbaden: Vieweg+Teubner, 2009. ISBN: 978-3-8348-0287-3.

[20] J. Ziegler, P. Bender, M. Schreiber, H. Lategahn, T. Strauss, C. Stiller, T. Dang, U. Franke, N. Appenrodt, C. G. Keller, et al. "Making Bertha Drive - An Autonomous Journey on a Historic Route." In: *IEEE Intelligent Transportation Systems Magazine* 6.2 (2014), pp. 8–20. ISSN: 1939-1390.

[21] J. Petit and S. E. Shladover. "Potential Cyberattacks on Automated Vehicles." In: *IEEE Transactions on Intelligent Transportation Systems* 16.2 (Apr. 2015), pp. 546–556. ISSN: 1524-9050.

[22] T. M. Gasser, C. Arzt, M. Ayoubi, A. Bartels, L. Bürkle, J. Eier, F. Flemisch, D. Häcker, T. Hesse, W. Huber, C. Lotz, M. Maurer, S. Ruth-Schumacher, J. Schwarz, and W. Vogt. "Rechtsfolgen zunehmender Fahrzeugautomatisierung." In: *Berichte der Bundesanstalt für Straßenwesen. Unterreihe Fahrzeugtechnik*. F 83. Jan. 2012. ISBN: 978-3-86918-189-9.

[23] *SAE J 3016 (2014), Taxonomy and Definitions for Terms Related to On-Road Motor Vehicle Automated Driving Systems, Society of Automotive Engineers International (SAE)*.

[24] C. Ress, D. Balzer, A. Bracht, S. Durekovic, and J. Löwenau. "Adasis Protocol for Advanced In-Vehicle Applications." In: *Proceedings of the 15th World Congress on Intelligent Transport Systems*. 2008, p. 7.

[25] S. Durekovic, A. Bracht, B. Raichle, M. Rauch, J. Requejo, D. Toropov, and A. Varchmin. *ADASIS v2 Protocol*. 2009.

[26] G. Araniti, C. Campolo, M. Condoluci, A. Iera, and A. Molinaro. "LTE for Vehicular Networking: A Survey." In: *IEEE Communications Magazine* 51.5 (May 2013), pp. 148–157. ISSN: 0163-6804.

[27] *ETSI ES 202 663 V1.1.0 (2009-11) Intelligent Transport Systems (ITS); European profile standard for the physical and medium access control layer of Intelligent Transport Systems operating in the 5 GHz frequency band*.

[28] G. R. Hiertz, D. Denteneer, L. Stibor, Y. Zang, X. P. Costa, and B. Walke. "The IEEE 802.11 Universe." In: *IEEE Communications Magazine* 48.1 (Jan. 2010), pp. 62–70. ISSN: 0163-6804.

[29] S. Gräfling, P. Mähönen, and J. Riihijärvi. "Performance Evaluation of IEEE 1609 WAVE and IEEE 802.11p for Vehicular Communications." In: *Proceedings of the 2nd International Conference on Ubiquitous and Future Networks (ICUFN)*. June 2010, pp. 344–348. ISBN: 978-1-4244-8087-6.

[30] *ETSI EN 302 636-1 V1.2.1 (2014-04) Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 1: Requirements*.

[31] *ETSI EN 302 636-2 V1.2.1 (2013-11) Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 2: Scenarios*.

[32] *ETSI EN 302 636-4-1 V1.2.1 (2014-05) Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 4: Geographical addressing and forwarding for point-to-point and point-to-multipoint communications; Sub-part 1: Media-Independent Functionality*.

[33] *ETSI EN 302 636-6-1 V1.2.1 (2014-05) Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 6: Internet Integration; Sub-part 1: Transmission of IPv6 Packets over GeoNetworking Protocols*.

[34] *ETSI EN 302 636-5-1 V1.2.1 (2014-08) Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 5: Transport Protocols; Sub-part 1: Basic Transport Protocol*.

[35] *ETSI EN 302 637-2 V1.3.2 (2014-11) Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service*.

[36] *IEEE 1609 Family of Standards for Wireless Access in Vehicular Environments (WAVE)*.

[37] *ETSI EN 302 637-3 V1.2.2 (2014-11) Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 3: Specifications of Decentralized Environmental Notification Basic Service*.

[38] *ETSI EN 302 895 V1.1.1 (2014-09) Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Local Dynamic Map (LDM)*.

[39] *ETSI TS 103 175 V1.1.1 (2015-06) Intelligent Transport Systems (ITS); Cross Layer DCC Management Entity for operation in the ITS G5A and ITS G5B medium*.

[40] *ETSI TS 102 687 V1.1.1 (2011-07) Intelligent Transport Systems (ITS); Decentralized Congestion Control Mechanisms for Intelligent Transport Systems operating in the 5 GHz range; Access layer part*.

[41] *ETSI TS 102 940 V1.1.1 (2012-06) Intelligent Transport Systems (ITS); Security; ITS communications security architecture and security management*.

[42] J. P. Snyder. *Map projections–A working manual*. Professional Paper 1395. US Government Printing Office, 1987. ISBN: 978-1782662228.

[43] K. Jo and M. Sunwoo. "Generation of a Precise Roadway Map for Autonomous Cars." In: *IEEE Transactions on Intelligent Transportation Systems* 15.3 (June 2014), pp. 925–937. ISSN: 1524-9050.

[44] L. Piegl and W. Tiller. *The NURBS book*. Springer Science & Business Media, 2012. ISBN: 978-3-540-61545-3.

[45] C. de Boor. "On calculating with B-splines." In: *Journal of Approximation Theory* 6.1 (1972), pp. 50–62. ISSN: 0021-9045.

[46] H. Prautzsch, W. Boehm, and M. Paluszny. *Bézier and B-spline techniques*. Springer Science & Business Media, 2013. ISBN: 978-3-642-07842-2.

[47] M. Quddus, W. Y. Ochieng, L. Zhao, and R. Noland. "A general map matching algorithm for transport telematics applications." English. In: *GPS Solutions* 7 (2003), pp. 157–167. ISSN: 1080-5370.

[48] R. E. Kalman. "A New Approach to Linear Filtering and Prediction Problems." In: *Transactions of the ASME - Journal of Basic Engineering* 82 (Series D) (1960), pp. 35–45.

[49] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking." In: *IEEE Transactions on Signal Processing* 50.2 (Feb. 2002), pp. 174–188. ISSN: 1053-587X.

[50] S. Rezaei and R. Sengupta. "Kalman Filter-Based Integration of DGPS and Vehicle Sensors for Localization." In: *IEEE Transactions on Control Systems Technology* 15.6 (Nov. 2007), pp. 1080–1088. ISSN: 1063-6536.

[51] D. Burgstahler, U. Lampe, N. Richerzhagen, and R. Steinmetz. "Push vs. Pull: An Energy Perspective." In: *Proceedings of the 6th IEEE International Conference on Service Oriented Computing & Applications (SOCA 2013)*. IEEE. Dec. 2013, pp. 190–193. ISBN: 978-1-4799-2701-2.

[52] D. Burgstahler, N. Richerzhagen, F. Englert, R. Hans, and R. Steinmetz. "Switching Push - Pull: An Energy Efficient Notification Approach." In: *Proceedings of the 3rd International Conference on Mobile Services (MS 2014)*. IEEE. June 2014, pp. 68–75. ISBN: 978-1-4799-5059-1.

[53] M. Bhide, P. Deolasee, A. Katkar, A. Panchbudhe, K. Ramamritham, and P. Shenoy. "Adaptive Push-Pull: Disseminating Dynamic Web Data." In: *IEEE Transactions on Computers* 51.6 (June 2002), pp. 652–668. ISSN: 0018-9340.

[54] P. T. Eugster, P. A. Felber, R. Guerraoui, and A.-M. Kermarrec. "The Many Faces of Publish/Subscribe." In: *ACM Comput. Surv.* 35.2 (June 2003), pp. 114–131. ISSN: 0360-0300.

[55] C. Maihöfer. "A Survey of Geocast Routing Protocols." In: *IEEE Communications Surveys Tutorials* 6.2 (Second 2004), pp. 32–42. ISSN: 1553-877X.

[56] C. L. Huang, Y. P. Fallah, R. Sengupta, and H. Krishnan. "Adaptive Intervehicle Communication Control for Cooperative Safety Systems." In: *IEEE Network* 24.1 (Jan. 2010), pp. 6–13. ISSN: 0890-8044.

[57] M. Sepulcre, J. Gozalvez, J. Harri, and H. Hartenstein. "Contextual Communications Congestion Control for Cooperative Vehicular Networks." In: *IEEE Transactions on Wireless Communications* 10.2 (Feb. 2011), pp. 385–389. ISSN: 1536-1276.

[58] G. Karagiannis, O. Altintas, E. Ekici, G. Heijenk, B. Jarupan, K. Lin, and T. Weil. "Vehicular Networking: A Survey and Tutorial on Requirements, Architectures, Challenges, Standards and Solutions." In: *IEEE Communications Surveys Tutorials* 13.4 (Fourth 2011), pp. 584–616. ISSN: 1553-877X.

[59] M. Armaghan, M. Fathy, and S. Yousefi. "Improving the Performance of Beacon Safety Message Dissemination in Vehicular Networks Using Kalman Filter Estimation." In: *Communication and Networking*. Springer, 2009, pp. 74–82. ISBN: 978-3-642-10844-0.

[60] H. Hartenstein and K. Laberteaux, eds. *VANET - Vehicular Applications and Inter-Networking Technologies*. Vol. 1. John Wiley & Sons Ltd., 2010, pp. 21–48, 81–105. ISBN: 978-0-470-74056-9.

[61] F. Bai, T. ElBatt, G. Holland, H. Krishnan, and V. Sadekar. "Towards Characterizing and Classifying Communication-based Automotive Applications from a Wireless Networking Perspective." In: *Proceedings of the 1st IEEE Workshop on Automotive Networking and Applications (AutoNet)*. Dec. 2006, pp. 1–25. ISBN: 1-4244-0356-1.

[62] Y. Toor, P. Muhlethaler, A. Laouiti, and A. D. L. Fortelle. "Vehicle Ad Hoc Networks: Applications and Related Technical Issues." In: *IEEE Communications Surveys Tutorials* 10.3 (2008), pp. 74–88. ISSN: 1553-877X.

[63] R. Steinmetz. *Kommunikation in Verteilten Systemen*. Springer, Mar. 1999. ISBN: 978-3-5406-5597-8.

[64] R. Steinmetz and K. Wehrle. *Peer-to-Peer Systems and Applications*. Springer, Sept. 2005. ISBN: 978-3-5402-9192-3.

[65] E. Uhlemann. "Connected-Vehicles Applications Are Emerging." In: *IEEE Vehicular Technology Magazine* 11.1 (Mar. 2016), pp. 25–96. ISSN: 1556-6072.

[66] C. Weiss. "V2X communication in Europe - From research projects towards standardization and field testing of vehicle communication technology." In: *Computer Networks* 55.14 (2011), pp. 3103–3119. ISSN: 1389-1286.

[67] S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti, and H. Zedan. "A comprehensive survey on vehicular Ad Hoc network." In: *Journal of Network and Computer Applications* 37 (2014), pp. 380–392. ISSN: 1084-8045.

[68] *ETSI TR 102 638 V1.1.1 (2009-06) Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Definitions*.

[69] *ETSI TS 101 539-1 V1.1.1 (2013-08) Intelligent Transport Systems (ITS); V2X Applications; Part 1: Road Hazard Signalling (RHS) application requirements specification*.

[70] *ETSI TS 101 539-3 V1.1.1 (2013-11) Intelligent Transport Systems (ITS); V2X Applications; Part 1: Longitudinal Collision Risk Warning (LCRW) application requirements specification*.

[71] S. Durekovic and N. Smith. "Architectures of Map-Supported ADAS." In: *Proceedings of the 2011 IEEE Intelligent Vehicles Symposium (IV)*. June 2011, pp. 207–211. ISBN: 978-1-4577-0891-6.

[72] S. Rezaei, R. Sengupta, H. Krishnan, X. Guan, and R. Bhatia. "Tracking the position of neighboring vehicles using wireless communications." In: *Transportation Research Part C: Emerging Technologies* 18.3 (2010), pp. 335–350. ISSN: 0968-090X.

[73] Y. P. Fallah, C. L. Huang, R. Sengupta, and H. Krishnan. "Analysis of Information Dissemination in Vehicular Ad-Hoc Networks With Application to Cooperative Vehicle Safety Systems." In: *IEEE Transactions on Vehicular Technology* 60.1 (Jan. 2011), pp. 233–247. ISSN: 0018-9545.

[74] A. Rostami, B. Cheng, G. Bansal, K. Sjöberg, M. Gruteser, and J. B. Kenney. "Stability Challenges and Enhancements for Vehicular Channel Congestion Control Approaches." In: *IEEE Transactions on Intelligent Transportation Systems* 17.10 (Oct. 2016), pp. 2935–2948. ISSN: 1524-9050.

[75] M. Slavik and I. Mahgoub. "Stochastic Broadcast for VANET." In: *Proceedings of the 7th IEEE Consumer Communications and Networking Conference*. Jan. 2010, pp. 1–5. ISBN: 978-1-4244-5176-0.

[76] S. Kühlmorgen, A. Festag, and G. Fettweis. "Impact of decentralized congestion control on contention-based forwarding in VANETs." In: *Proceedings of the 17th IEEE International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*. June 2016, pp. 1–7. ISBN: 978-1-5090-2185-7.

[77] M. Torrent-Moreno, J. Mittag, P. Santi, and H. Hartenstein. "Vehicle-to-Vehicle Communication: Fair Transmit Power Control for Safety-Critical Information." In: *IEEE Transactions on Vehicular Technology* 58.7 (Sept. 2009), pp. 3684–3703. ISSN: 0018-9545.

[78] R. Baldessari, D. Scanferla, L. Le, W. Zhang, and A. Festag. "Joining Forces for VANETs: A Combined Transmit Power and Rate Control Algorithm." In: *Proceedings of the 7th International Workshop on Intelligent Transportation (WIT)*. Hamburg, 2010.

[79] T. Tielert, D. Jiang, Q. Chen, L. Delgrossi, and H. Hartenstein. "Design Methodology and Evaluation of Rate Adaptation Based Congestion Control for Vehicle Safety Communications." In: *Proceedings of the 2011 IEEE Vehicular Networking Conference (VNC)*. Nov. 2011, pp. 116–123. ISBN: 978-1-4673-0047-6.

[80] J. B. Kenney, G. Bansal, and C. E. Rohrs. "LIMERIC: A Linear Message Rate Control Algorithm for Vehicular DSRC Systems." In: *Proceedings of the 8th ACM International Workshop on Vehicular Inter-networking*. VANET '11. ACM, Sept. 2011, pp. 21–30. ISBN: 978-1-4503-0869-4.

[81] G. Bansal, B. Cheng, A. Rostami, K. Sjoberg, J. B. Kenney, and M. Gruteser. "Comparing LIMERIC and DCC approaches for VANET channel congestion control." In: *Proceedings of the 6th IEEE International Symposium on Wireless Vehicular Communications (WiVeC)*. Sept. 2014, pp. 1–7. ISBN: 978-1-4799-4452-1.

[82] A. Rostami, B. Cheng, G. Bansal, K. Sjöberg, M. Gruteser, and J. B. Kenney. "Stability Challenges and Enhancements for Vehicular Channel Congestion Control Approaches." In: *IEEE Transactions on Intelligent Transportation Systems* 17.10 (Oct. 2016), pp. 2935–2948. ISSN: 1524-9050.

[83] B. Cheng, A. Rostami, M. Gruteser, J. B. Kenney, G. Bansal, and K. Sjoberg. "Performance Evaluation of a Mixed Vehicular Network with CAM-DCC and LIMERIC Vehicles." In: *Proceedings of the 16th IEEE International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*. June 2015, pp. 1–6. ISBN: 978-1-4799-8461-9.

[84] G. Bansal and J. B. Kenney. "Controlling Congestion in Safety-Message Transmissions: A Philosophy for Vehicular DSRC Systems." In: *IEEE Vehicular Technology Magazine* 8.4 (Dec. 2013), pp. 20–26. ISSN: 1556-6072.

[85] E. Egea-Lopez and P. Pavon-Mariño. "Distributed and Fair Beaconing Rate Adaptation for Congestion Control in Vehicular Networks." In: *IEEE Transactions on Mobile Computing* 15.12 (Dec. 2016), pp. 3028–3041. ISSN: 1536-1233.

[86] J. Jose, C. Li, X. Wu, L. Ying, and K. Zhu. "Distributed Rate and Power Control in DSRC." In: *Proceedings of the 2015 IEEE International Symposium on Information Theory (ISIT)*. June 2015, pp. 2822–2826. ISBN: 978-1-4673-7704-1.

[87] H. Piao, Y. Park, B. Kim, and H. Kim. "Safety Beaconing Rate Control Based on Vehicle Counting in WAVE." In: *Proceedings of the 2015 IEEE Intelligent Vehicles Symposium (IV)*. June 2015, pp. 1361–1366. ISBN: 978-1-4673-7266-4.

[88] B. Aygun, M. Boban, and A. M. Wyglinski. "ECPR: Environment-and context-aware combined power and rate distributed congestion control for vehicular communications." In: *Computer Communications* 93 (2016), pp. 3–16. ISSN: 0140-3664.

[89] N. Taherkhani and S. Pierre. "Centralized and Localized Data Congestion Control Strategy for Vehicular Ad Hoc Networks Using a Machine Learning Clustering Algorithm." In: *IEEE Transactions on Intelligent Transportation Systems* 17.11 (Nov. 2016), pp. 3275–3285. ISSN: 1524-9050.

[90] S. Rezaei, R. Sengupta, and H. Krishnan. "Reducing the Communication Required By DSRC-Based Vehicle Safety Systems." In: *Proceedings of the IEEE Intelligent Transportation Systems Conference*. Sept. 2007, pp. 361–366.

[91] M. Segata, F. Dressler, and R. L. Cigno. "Jerk Beaconing: A Dynamic Approach to Platooning." In: *Proceedings of the 2015 IEEE Vehicular Networking Conference (VNC)*. Dec. 2015, pp. 135–142. ISBN: 978-1-4673-9411-6.

[92] C. L. Huang, Y. P. Fallah, R. Sengupta, and H. Krishnan. "Information Dissemination Control for Cooperative Active Safety Applications in Vehicular Ad-Hoc Networks." In: *Proceedings of the 2009 IEEE Global Telecommunications Conference (GLOBECOM)*. Nov. 2009, pp. 1–6. ISBN: 978-1-4244-4148-8.

[93] C. L. Huang, Y. P. Fallah, R. Sengupta, and H. Krishnan. "Intervehicle Transmission Rate Control for Cooperative Active Safety System." In: *IEEE Transactions on Intelligent Transportation Systems* 12.3 (Sept. 2011), pp. 645–658. ISSN: 1524-9050.

[94] G. Bansal, H. Lu, J. B. Kenney, and C. Poellabauer. "EMBARC: Error Model Based Adaptive Rate Control for Vehicle-to-vehicle Communications." In: *Proceeding of the 10th ACM International Workshop on Vehicular Inter-networking, Systems, and Applications*. VANET '13. Taipei, Taiwan: ACM, June 2013, pp. 41–50. ISBN: 978-1-4503-2073-3.

[95] M. Sepulcre, J. Gozalvez, O. Altintas, and H. Kremo. "Adaptive Beaconing for Congestion and Awareness Control in Vehicular Networks." In: *Proceedings of the 2014 IEEE Vehicular Networking Conference (VNC)*. Dec. 2014, pp. 81–88. ISBN: 978-1-4799-7660-7.

[96] D. Burgstahler, T. Meuser, U. Lampe, D. Böhnstedt, and R. Steinmetz. "ProbSense.KOM: A Probabilistic Sensing Approach for Gathering Vehicular Sensed Data." In: *Proceedings of the 3rd International Conference on Mobile Services (MS 2016)*. IEEE. June 2016, pp. 9–16. ISBN: 978-1-5090-2625-8.

[97] N. D. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. T. Campbell. "A Survey of Mobile Phone Sensing." In: *IEEE Communications Magazine* 48.9 (2010), pp. 140–150. ISSN: 0163-6804.

[98] H. Ma, D. Zhao, and P. Yuan. "Opportunities in Mobile Crowd Sensing." In: *IEEE Communications Magazine* 52.8 (2014), pp. 29–35. ISSN: 0163-6804.

[99] N. D. Lane, S. B. Eisenman, M. Musolesi, E. Miluzzo, and A. T. Campbell. "Urban Sensing Systems: Opportunistic or Participatory?" In: *Proceedings of the 9th Workshop on Mobile Computing Systems and Applications*. Feb. 2008. ISBN: 978-1-60558-118-7.

[100] K. Shilton. "Four Billion Little Brothers?: Privacy, Mobile Phones, and Ubiquitous Data Collection." In: *Communications of the ACM* 52.11 (2009), pp. 48–53. ISSN: 0001-0782.

[101] M. Shin, C. Cornelius, D. Peebles, A. Kapadia, D. Kotz, and N. Triandopoulos. "AnonySense: A System for Anonymous Opportunistic Sensing." In: *Pervasive and Mobile Computing* 7.1 (2011), pp. 16–30. ISSN: 1574-1192.

[102] A. Hossain, P. Biswas, and S. Chakrabarti. "Sensing Models and its Impact on Network Coverage in Wireless Sensor Network." In: *Proceedings of the 3rd International Conference on Industrial and Information Systems*. Mar. 2008. ISBN: 978-1-4244-2806-9.

[103] K. Tischler and B. Hummel. "Enhanced Environmental Perception by Inter-Vehicle Data Exchange." In: *Proceedings of the IEEE Intelligent Vehicles Symposium 2005*. Sept. 2005. ISBN: 0-7803-8961-1.

[104] U. Lee, E. Magistretti, B. Zhou, M. Gerla, P. Bellavista, and A. Corradi. "Efficient Data Harvesting in Mobile Sensor Platforms." In: *Proceedings of the 4th IEEE International Conference on Pervasive Computing and Communications Workshops*. Mar. 2006. ISBN: 0-7695-2520-2.

[105] S. Vodopivec, J. Bešter, and A. Kos. "A Survey on Clustering Algorithms for Vehicular Ad-Hoc Networks." In: *Proceedings of the 35th International Conference on Telecommunications and Signal Processing*. July 2012. ISBN: 978-1-4673-1118-2.

[106] C. R. Lin and M. Gerla. "Adaptive Clustering for Mobile Wireless Networks." In: *Selected Areas in Communications, IEEE Journal on* 15.7 (1997), pp. 1265–1275. ISSN: 0733-8716.

[107] M. Gerla and J. T.-C. Tsai. "Multicluster, Mobile, Multimedia Radio Network." In: *Wireless Networks* 1.3 (1995), pp. 255–265. ISSN: 1572-8196.

[108] M. Chatterjee, S. K. Das, and D. Turgut. "WCA: A Weighted Clustering Algorithm for Mobile Ad Hoc Networks." In: *Cluster Computing* 5.2 (2002), pp. 193–204. ISSN: 1573-7543.

[109] P. Basu, N. Khan, and T. D. Little. "A Mobility Based Metric for Clustering in Mobile Ad Hoc Networks." In: *Proceedings of the 2001 International Conference on Distributed Computing Systems Workshop*. Apr. 2001. ISBN: 0-7695-1080-9.

[110] Y. Zhang, J. M. Ng, and C. P. Low. "A Distributed Group Mobility Adaptive Clustering Algorithm for Mobile Ad Hoc Networks." In: *Computer Communications* 32.1 (2009), pp. 189–202. ISSN: 0140-3664.

[111] M. M. C. Morales, C. S. Hong, and Y.-C. Bang. "An Adaptable Mobility-Aware Clustering Algorithm in Vehicular Networks." In: *Proceedings of the 13th Asia-Pacific Network Operations and Management Symposium*. Sept. 2011. ISBN: 978-1-4577-1670-6.

[112] CONVERGE Project. *Deliverable D6 Final Assessment: Appendix 8.3 Results of Simulation*. Accessed February 2017, `http://www.converge-online.de/doc/download/D6-AP8-Final-Assessment.pdf`. 2015.

[113] M. Miskowicz. "Send-on-Delta Concept: An Event-Based Data Reporting Strategy." In: *Sensors* 6.1 (2006), pp. 49–63. ISSN: 1424-8220.

[114] Y. S. Suh. "Send-on-Delta Sensor Data Transmission With a Linear Predictor." In: *Sensors* 7.4 (2007), pp. 537–547. ISSN: 1424-8220.

[115] D. Chu, A. Deshpande, J. M. Hellerstein, and W. Hong. "Approximate Data Collection in Sensor Networks Using Probabilistic Models." In: *Proceedings of the 22nd International Conference on Data Engineering*. Apr. 2006. ISBN: 0-7695-2570-9.

[116] A. Deshpande, C. Guestrin, S. R. Madden, J. M. Hellerstein, and W. Hong. "Model-Driven Data Acquisition in Sensor Networks." In: *Proceedings of the Thirtieth International Conference on Very Large Data Bases*. Aug. 2004. ISBN: 0-12-088469-0.

[117] B. Hull, V. Bychkovsky, Y. Zhang, K. Chen, M. Goraczko, A. Miu, E. Shih, H. Balakrishnan, and S. Madden. "CarTel: A Distributed Mobile Sensor Computing System." In: *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems (SenSys '06)*. ACM, Oct. 2006, pp. 125–138. ISBN: 1-59593-343-3.

[118] S. Li, L. Da Xu, and X. Wang. "Compressed Sensing Signal and Data Acquisition in Wireless Sensor Networks and Internet of Things." In: *IEEE Transactions on Industrial Informatics* 9.4 (2013), pp. 2177–2186. ISSN: 1941-0050.

[119] V. Blervaque, K. Mezger, L. Beuk, and J. Loewenau. "ADAS Horizon - How Digital Maps can contribute to Road Safety." In: *Advanced Microsystems for Automotive Applications 2006*. Ed. by J. Valldorf and W. Gessner. Springer, 2006, pp. 427–436. ISBN: 978-3-540-33410-1.

[120] C. Ress, A. Etemad, D. Kuck, and M. Boerger. "Electronic Horizon - Supporting ADAS Applications with Predictive Map Data." In: *Proceedings of the 15th World Congress on Intelligent Transport Systems*. London, Oct. 2006.

[121] J. Ludwig. "Electronic horizon: Flexible implementation of predictive driver assistance features." In: *15. Internationales Stuttgarter Symposium: Automobil- und Motorentechnik*. Ed. by M. Bargende, H.-C. Reuss, and J. Wiedemann. Springer, 2015, pp. 1049–1062. ISBN: 978-3-658-08844-6.

[122] P. Engel, W. Balkema, and A. Varchmin. "Verfahren und Anordnung zum Bestimmen eines am ehesten wahrscheinlichen Fahrpfads eines Fahrzeugs." Patent DE102011078946 A1. Jan. 17, 2013.

[123] B. Thomas, J. Lowenau, S. Durekovic, and H. U. Otto. "The ActMAP - FeedMAP framework - A basis for in-vehicle ADAS application improvement." In: *Proceedings of the 2008 IEEE Intelligent Vehicles Symposium*. June 2008, pp. 263–268. ISBN: 978-1-4244-2568-6.

[124] Y. Horita and R. S. Schwartz. "Extended Electronic Horizon for Automated Driving." In: *Proceedings of the 14th International Conference on ITS Telecommunications (ITST)*. Dec. 2015, pp. 32–36. ISBN: 978-1-4673-9382-9.

[125] L. Zhang and S. Valaee. "Congestion Control for Vehicular Networks With Safety-Awareness." In: *IEEE/ACM Transactions on Networking* 24.6 (Dec. 2016), pp. 3290–3299. ISSN: 1063-6692.

[126] A. Autolitano, C. Campolo, A. Molinaro, R. M. Scopigno, and A. Vesco. "An Insight into Decentralized Congestion Control Techniques for VANETs from ETSI TS 102 687 V1.1.1." In: *Proceedings of the 2013 IFIP Wireless Days (WD)*. Nov. 2013, pp. 1–6. ISBN: 978-1-4799-0543-0.

[127] S. E. Shladover and S.-K. Tan. "Analysis of Vehicle Positioning Accuracy Requirements for Communication-Based Cooperative Collision Warning." In: *Journal of Intelligent Transportation Systems* 10.3 (2006), pp. 131–140.

[128] H. Pu. "Using eHorizon to Enhance Camera-Based Environmental Perception for Advanced Driver Assistance Systems and Automated Driving." In: *Advanced Microsystems for Automotive Applications 2016: Smart Systems for the Automobile of the Future*. Ed. by T. Schulze, B. Müller, and G. Meyer. Springer, 2016, pp. 103–112. ISBN: 978-3-319-44766-7.

[129] C. Campolo, A. Molinaro, and R. Scopigno, eds. *Vehicular ad hoc Networks: Standards, Solutions, and Research*. Springer, 2015, pp. 525–544. ISBN: 978-3-319-15497-1.

[130] D. Burgstahler, M. Möbus, T. Meuser, D. Böhnstedt, and R. Steinmetz. "A Categorization Scheme for Information Demands of Future Connected ADAS (accepted for publication)." In: *GMM-Fachbericht-AmE 2017–Automotive meets Electronics*. VDE VERLAG GmbH, Mar. 2017.

[131] E. Uhlemann. "Introducing Connected Vehicles." In: *IEEE Vehicular Technology Magazine* 10.1 (Mar. 2015), pp. 23–31. ISSN: 1556-6072.

[132] D. Jiang, Q. Chen, and L. Delgrossi. "Optimal Data Rate Selection for Vehicle Safety Communications." In: *Proceedings of the Fifth ACM International Workshop on VehiculAr Inter-NETworking (VANET '08)*. ACM, Sept. 2008, pp. 30–38. ISBN: 978-1-60558-191-0.

[133] *ETSI TS 102 792 V1.2.1 (2015-06) Intelligent Transport Systems (ITS); Mitigation techniques to avoid interference between European CEN Dedicated Short Range Communication (CEN DSRC) equipment and Intelligent Transport Systems (ITS) operating in the 5 GHz frequency range*.

[134] R. Nagel and S. Eichler. "Efficient and Realistic Mobility and Channel Modeling for VANET Scenarios using OMNeT++ and INET-Framework." In: *Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems & Workshops*. Simutools '08. ICST, Mar. 2008, pp. 1–8. ISBN: 978-963-9799-20-2.

[135] C. Sommer, D. Eckhoff, R. German, and F. Dressler. "A Computationally Inexpensive Empirical Model of IEEE 802.11p Radio Shadowing in Urban Environments." In: *Proceedings of the 8th International Conference on Wireless On-Demand Network Systems and Services (WONS)*. Jan. 2011, pp. 84–90. ISBN: 978-1-61284-188-5.

[136] M. Boban, J. Barros, and O. K. Tonguz. "Geometry-Based Vehicle-to-Vehicle Channel Modeling for Large-Scale Simulation." In: *IEEE Transactions on Vehicular Technology* 63.9 (Nov. 2014), pp. 4146–4164. ISSN: 0018-9545.

[137] G. Durgin, T. S. Rappaport, and H. Xu. "Measurements and Models for Radio Path Loss and Penetration Loss In and Around Homes and Trees at 5.85 GHz." In: *IEEE Transactions on Communications* 46.11 (Nov. 1998), pp. 1484–1496. ISSN: 0090-6778.

[138] K. Mahler, P. Paschalidis, A. Kortke, M. Peter, and W. Keusgen. "Realistic IEEE 802.11p Transmission Simulations Based on Channel Sounder Measurement Data." In: *Proceedings of the 2013 78th IEEE Vehicular Technology Conference (VTC2013-Fall)*. Sept. 2013, pp. 1–5.

[139] R. Meireles, M. Boban, P. Steenkiste, O. Tonguz, and J. Barros. "Experimental Study on the Impact of Vehicular Obstructions in VANETs." In: *Proceedings of the 2010 IEEE Vehicular Networking Conference (VNC)*. Dec. 2010, pp. 338–345.

[140] P. Paschalidis, K. Mahler, A. Kortke, M. Wisotzki, M. Peter, and W. Keusgen. "2 X 2 MIMO Measurements of the Wideband Car-to-Car Channel at 5.7 GHz on Urban Street Intersections." In: *Proceedings of the 2011 IEEE 74th Vehicular Technology Conference (VTC2011-Fall)*. Sept. 2011, pp. 1–5.

[141] K. Mahler, P. Paschalidis, M. Wisotzki, A. Kortke, and W. Keusgen. "Evaluation of Vehicular Communication Performance at Street Intersections." In: *Proceedings of the 2014 IEEE 80th Vehicular Technology Conference (VTC2014-Fall)*. Sept. 2014, pp. 1–5.

[142] M. Sepulcre, J. Mittag, P. Santi, H. Hartenstein, and J. Gozalvez. "Congestion and Awareness Control in Cooperative Vehicular Systems." In: *Proceedings of the IEEE* 99.7 (July 2011), pp. 1260–1279. ISSN: 0018-9219.

[143] Y. P. Fallah, C. Huang, R. Sengupta, and H. Krishnan. "Congestion Control Based on Channel Occupancy in Vehicular Broadcast Networks." In: *Proceedings of the 2010 IEEE 72nd Vehicular Technology Conference (VTC2011-Fall)*. Sept. 2010, pp. 1–5. ISBN: 978-1-4244-8328-0.

[144] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos. "Context Aware Computing for The Internet of Things: A Survey." In: *IEEE Communications Surveys Tutorials* 16.1 (2014), pp. 414–454. ISSN: 1553-877X.

[145] T. Meuser, D. Burgstahler, T. Rückelt, D. Böhnstedt, and R. Steinmetz. "Hybrid-ProbSense.KOM: Probabilistic Sensing with Hybrid Communication for Gathering Vehicular Sensed Data (accepted for publication)." In: *GMM-Fachbericht-AmE 2017–Automotive meets Electronics*. VDE VERLAG GmbH, Mar. 2017.

[146] S. M. Stigler. "Gauss and the Invention of Least Squares." In: *The Annals of Statistics* 9.3 (1981), pp. 465–474. ISSN: 00905364.

[147] D. Burgstahler, A. Xhoga, C. Peusens, M. Möbus, D. Böhnstedt, and R. Steinmetz. "RemoteHorizon.KOM: Dynamic Cloud-based eHorizon." In: *GMM-Fachbericht-AmE 2016–Automotive meets Electronics*. VDE VERLAG GmbH, Mar. 2016, pp. 46–51. ISBN: 978-3-8007-4167-0.

[148] HERE. *Vehicle Sensor Data Cloud Ingestion Interface Specification v2.0.2*. Accessed February 2017, `https://lts.cms.here.com/static-cloud-content/Company_Site/2015_06/Vehicle_Sensor_Data_Cloud_Ingestion_Interface_Specification.pdf`. 2015.

[149] R. Ratasuk, A. Prasad, Z. Li, A. Ghosh, and M. A. Uusitalo. "Recent Advancements in M2M Communications in 4G Networks and Evolution Towards 5G." In: *2015 18th International Conference on Intelligence in Next Generation Networks*. Feb. 2015, pp. 52–57. ISBN: 978-1-4799-1866-9.

[150] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker. "Recent Development and Applications of SUMO - Simulation of Urban MObility." In: *International Journal On Advances in Systems and Measurements* 5.3&4 (Dec. 2012), pp. 128–138. ISSN: 1942-261x.

[151] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz. "SUMO - Simulation of Urban MObility: An Overview." In: *Proceedings the third International Conference on Advances in System Simulation (SIMUL 2011)*. Sept. 2011. ISBN: 978-1-61208-169-4.

[152] C. Varschen and P. Wagner. "Mikroskopische Modellierung der Personenverkehrsnachfrage auf Basis von Zeitverwendungstagebüchern." In: *Integrierte Mikro-Simulation von Raum- und Verkehrsentwicklung. Theorie, Konzepte, Modelle, Praxis* 81 (2006), pp. 63–69. ISSN: 0344-9645.

[153] L. Codeca, R. Frank, and T. Engel. "Luxembourg SUMO Traffic (LuST) Scenario: 24 hours of mobility for vehicular networking research." In: *Proceedings of the 2015 IEEE Vehicular Networking Conference (VNC)*. Dec. 2015, pp. 1–8. ISBN: 978-1-4673-9411-6.

[154] R. Kirk. *Statistics: An Introduction*. Wadsworth Publishing Company Inc., 2006. ISBN: 978-0534564780.

[155] D. Burgstahler, C. Peusens, D. Boehnstedt, and R. Steinmetz. "Horizon.KOM: A First Step Towards an Open Vehicular Horizon Provider." In: *Proceedings of the 2nd International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS)*. Apr. 2016, pp. 79–84. ISBN: 978-989-758-185-4.

[156] A. Sumaray and S. K. Makki. "A Comparison of Data Serialization Formats for Optimal Efficiency on a Mobile Platform." In: *Proceedings of the 6th International Conference on Ubiquitous Information Management and Communication (ICUIMC)*. Feb. 2012, pp. 1–6. ISBN: 978-1-4503-1172-4.

[157] *ETSI TR 101 613 V1.1.1 (2015-09) Intelligent Transport Systems (ITS); Cross Layer DCC Management Entity for operation in the ITS G5A and ITS G5B medium; Validation set-up and results*.

[158] X. Shen, X. Cheng, R. Zhang, B. Jiao, and Y. Yang. "Distributed Congestion Control Approaches for the IEEE 802.11p Vehicular Networks." In: *IEEE Intelligent Transportation Systems Magazine* 5.4 (2013), pp. 50–61. ISSN: 1939-1390.

[159] S. Eichler. "Performance Evaluation of the IEEE 802.11p WAVE Communication Standard." In: *Proceedings of the 2007 IEEE 66th Vehicular Technology Conference (VTC2007-Fall)*. Sept. 2007, pp. 2199–2203.

[160] J. Rankin. "GPS and differential GPS: An error model for sensor simulation." In: *Proceedings of the IEEE Position Location and Navigation Symposium*. Apr. 1994, pp. 260–266. ISBN: 0-7803-1435-2.

[161] K. Bösche, T. Sellam, H. Pirk, R. Beier, P. Mieth, and S. Manegold. "Scalable Generation of Synthetic GPS Traces with Real-Life Data Characteristics." In: *Selected Topics in Performance Evaluation and Benchmarking: 4th TPC Technology Conference, TPCTC 2012, Istanbul, Turkey, August 27, 2012, Revised Selected Papers*. Ed. by R. Nambiar and M. Poess. Springer, 2013, pp. 140–155. ISBN: 978-3-642-36727-4.

[162] W. Mansfeld. *Satellitenortung und Navigation: Grundlagen und Anwendung globaler Satellitennavigationssysteme*. Springer-Verlag, 2013, pp. 202–204. ISBN: 978-3-8348-0611-6.

[163] *ETSI TR 102 962 V1.1.1 (2012-02) Intelligent Transport Systems (ITS); Framework for Public Mobile Networks in Cooperative ITS (C-ITS)*.

[164] Y. C. Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young. "Mobile Edge Computing - A key technology towards 5G." In: *ETSI White Paper*. Vol. 11. 2015. ISBN: 979-10-92620-08-5.

[165] K. David and N. Jefferies. "Wireless Visions: A Look to the Future by the Fellows of the WWRF." In: *IEEE Vehicular Technology Magazine* 7.4 (Dec. 2012), pp. 26–36. ISSN: 1556-6072.

LIST OF ACRONYMS

ACC	Adaptive Cruise Control
ADAS	Advanced Driver Assistance System
ADASIS	Advanced Driver Assistance Systems Interface Specification
AIFS	Arbitration Inter-Frame Spacing
BSM	Basic Safety Message
BTP	Basic Transport Protocol
CA	Cooperative Awareness
CAM	Cooperative Awareness Message
CBR	Channel Busy Ratio
CS	Compressed Sensing
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CDN	Content Delivery Network
C-ITS	Cooperative Intelligent Transport System
CPM	Cooperative Perception Message
CSMA	Carrier Sense Multiple Access
DCC	Decentralized Congestion Control
DENM	Decentralized Environment Notification Message
DIFS	Distributed Coordination Function Interframe Space
DSRC	Dedicated Short Range Communication
DGPS	Differential Global Positioning System
eHorizon	electronic Horizon
eMBMS	evolved Multimedia Broadcast Multicast Service
EIRP	Equivalent Isotropically Radiated Power
ESC	Electronic Stability Control
ETSI	European Telecommunications Standards Institute
FSPL	Free Space Path Loss
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
HTTP	Hypertext Transfer Protocol
IoT	Internet of Things
ITS	Intelligent Transportation System
ITT	Inter Transmit Time
JSON	JavaScript Object Notation
LBS	Location Based Services
LDM	Local Dynamic Map
LIDAR	LIght Detection And Ranging

LTE	Long Term Evolution
LTE-A	Long Term Evolution Advanced
MAC	Medium Access Control
MBMS	Multimedia Broadcast Multicast Service
eMBMS	evolved Multimedia Broadcast Multicast Service
MCS	Mobile Crowd Sourcing
MPP	Most Probable Path
MQTT	Message Queuing Telemetry Transport
OSM	Open Street Map
PER	Packet Error Rate
PKI	Public Key Infrastructure
PoI	Point of Interest
POTI	Position and Time management
QoS	Quality of Service
RADAR	RAdio Detection And Ranging
RSSI	Received Signal Strength Indication
RSU	Road Side Unit
SNR	Signal to Noise Ratio
SPAT	Signal Phase And Timing Message
SUMO	Simulation of Urban MObility
TA	Trusted Authority
TPC	Transmit Power Control
UMTS	Universal Mobile Telecommunications System
UTM	Universal Transverse Mercator
VANET	Vehicular Ad-hoc NETwork
VDP	Vehicle Data Provider
V2I	Vehicle to Infrastructure
V2V	Vehicle to Vehicle
V2X	Vehicle to everything
VIN	Vehicle Identification Number
WAVE	Wireless Access in Vehicular Environments
WGS84	World Geodetic Datum 1984
MANET	Mobile Ad-hoc Network
TLS	Transport Layer Security
XML	Extensible Markup Language

A

APPENDIX

A.1 USE CASE ITEMIZATION

As explained in Chapter 4, we have derived three zones of information demands from the use cases described in the ETSI specifications and standards as well as relevant research publications [68, 69, 70, 38, 61, 66, 129]. We have named these three zones *safety zone*, *awareness zone* and *information zone*. As mentioned before, the distinction of these zones belongs to the respective tolerated latency between the occurrence of an event and the point in time, the information has to be processed at the receiver side. Each of these zones contains of a set of related use cases. We have extracted these use cases out of literature and categorized them into subgroups. Each subgroup is assigned an ETSI application class, the ETSI message types, and the common goal.

A.1.1 *Safety Zone Use Cases*

Use Cases of the *safety zone* are listed in Figure 67. These use cases have a high safety impact and the maximum latency between the occurrence of an event and the point in time the information has to be processed at the receiver side is below 2-5 seconds. In general, for these use cases the respective application has to react automatically, because of the low reaction time. For use case realizations, mobile ad-hoc communication is the appropriate technology. This is due to the high latency demands and the fact that the availability of cellular networks cannot be guaranteed within the short communication window. The *safety zone* has two main subcategories, namely collision prevention and collision risk warning. The respective use case examples are explained in the following.

Collision Prevention
The main purpose of this subcategory is to increase driving safety. Focus is here the direct prevention of an imminent collision. In ETSI documents it is categorized as active road safety. These use cases are mostly covered in the EU by the already specified DENM and in the US by the BSM.

- **Merging traffic turn warning:** A vehicle merges the main road at a T-junction. Vehicles on the main road have to be informed about the presence, position and movement of the incoming vehicle to avoid lateral collision. If no line of sight between the vehicles is possible, e. g., obstructed by buildings, a RSU should be installed to relay messages [68].

- **Across traffic turn warning:** A vehicle drives along the main road and is intending to turn left, i. e., crossing the opposing traffic. In a country of driving left, this turn is accordingly to the right. By the use of communication technology, approaching vehicles are informed that the transmitting vehicle is intending to

turn left, across opposing traffic. A lateral collision should be prevented. The indication to turn does not authorize the respective vehicle. If the vehicle regardless starts the turn maneuver, i. e., violates the right of way, an emergency break or evasive trajectory can be started as early as possible to prevent an accident, or at least to mitigate the collision.

- **Cooperative forward collision warning:** Several vehicles drive consecutively on a road. One vehicle has an accident or has to execute an emergency break. Information about event type and position is broadcasted by the first most vehicle. Following vehicles relay the message to extend notification range. A longitudinal collision should be prevented. Vehicles that receive the message can react accordingly and instantaneously execute an emergency break or evasive trajectory.

- **Cooperative glare reduction:** Vehicles are driving towards each other at night, e. g., on a country road. Both vehicles are able to automatically switch from high-beams to low-beams, by detecting each other, enabled by the exchange of information about presence, position, speed and driving direction. A frontal collision should be prevented.

- **Signal violation warning:** A vehicle is violating a traffic signal, e. g., driving over a red traffic light, and deductively increases an accident risk. The detection can e. g., be realized by the traffic light itself, equipped with sensors and a RSU. A collision of affected vehicles should be prevented. Respective information is broadcasted to affected vehicles in the surroundings.

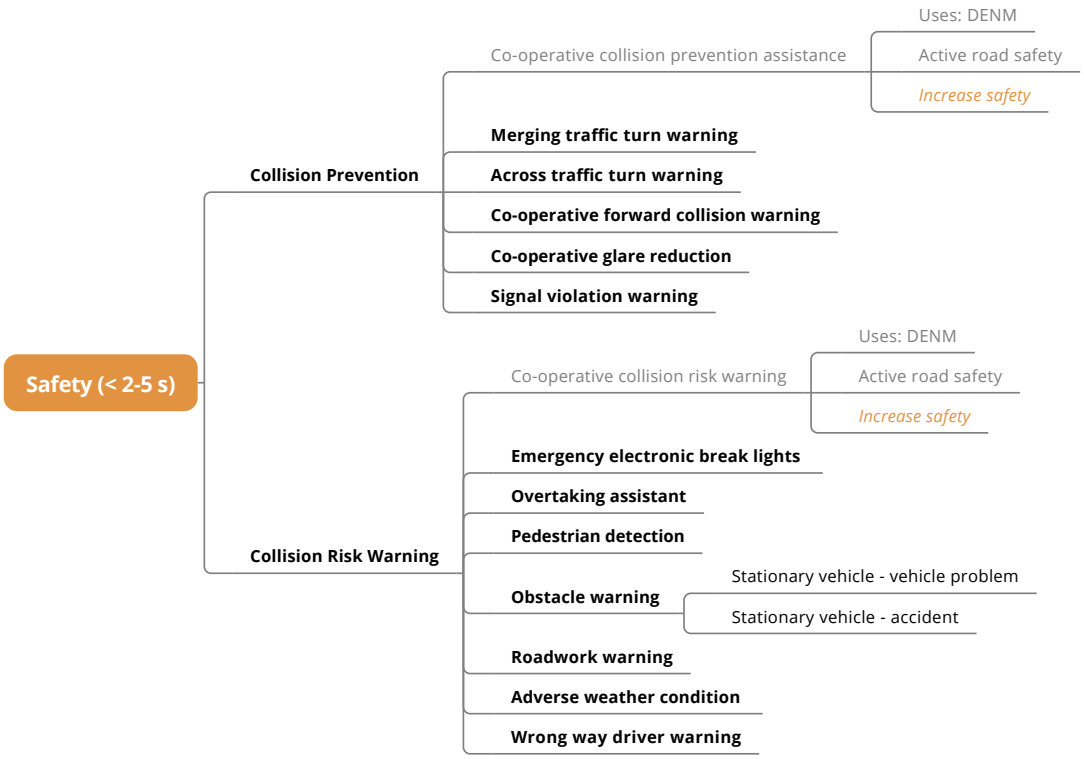

Figure 67: Itemization of the safety zone related use cases.

Collision risk warning
The main purpose of this subcategory is to increase driving safety. Focus is here the warning of a possible collision risk in direct proximity of the ego vehicle. In ETSI documents it is categorized as active road safety. These use cases are mostly covered in the EU by the already specified DENM and in the US by the BSM.

- **Emergency electronic break lights:** A vehicle has to execute hard breaking. This triggers the broadcast of a respective message to the following vehicles, to inform about a sudden traffic slowdown. A longitudinal collision should be prevented.

- **Overtaking assistant:** A vehicle broadcasts maneuver status information, while performing an overtaking action. Affected vehicles in the surroundings are aware of the situation. This can secure the situation and reduce the risk of an accident, especially in boundary conditions. A special and also safety relevant situation could be a lane change on a autobahn. A vehicle wants to change to the left lane for overtaking a vehicle driving ahead. On the left lane is another vehicle approaching, driving at high velocity. This situation results in a risk of longitudinal collision. In a country of driving left, the lane change is accordingly to the right. Broadcast of position beacons allows to detect the situation. The approaching vehicle can slow down and the other vehicle neglect the lane change to avoid a collision. Another critical situation could be two vehicles approaching each other, which have both started an overtaking maneuver. Broadcast of position beacons allows to detect the collision risk. This is a typical motorcycle accident situation [70].

- **Pedestrian detection:** An ITS station, i. e., a vehicle or a sensor equipped RSU, detects a vulnerable road user, e. g., a pedestrian, and informs other road users in the surroundings by a respective broadcast message. The potential dangerous situation should be mitigated and a collision or accident be prevented.

- **Obstacle warning:** An ITS station, i. e., a vehicle or a sensor equipped RSU, detects an obstacle on the road. An obstacle could be a stationary vehicle on the road, caused by a breakdown or an accident. A longitudinal collision should be prevented. Other road users in the surroundings are informed by a respective broadcast message.

- **Roadwork warning:** The end of a roadwork is equipped with a RSU that broadcasts information about the roadwork and related restrictions. Typically traffic guidance is changed around roadworks and single lanes might be closed. An accident or longitudinal collision should be prevented. If the roadwork itself is not equipped with a RSU, the end of the roadwork might be detected by a vehicle and respective information is broadcasted to road users in the surroundings.

- **Adverse weather condition:** An ITS station, i. e., a vehicle or a sensor equipped RSU, detects an adverse weather condition, e. g., strong wind gust on a bridge, black ice, oil, low adhesion or snow on the road. A respective warning message is broadcasted to inform other road users in the surroundings about the hazardous situation. An accident should be prevented.

- **Wrong way driver warning:** A vehicle is driving against permitted driving direction. This might be detected by sensors or by analysis of received position beacons, i. e., CAM. A warning message, i. e., DENM, is broadcasted for a limited time and relayed towards the affected region, i. e., towards vehicles approaching the wrong way driving vehicle. Frontal collisions should limited as much as possible.

A.1.2 *Awareness Zone Use Cases*

Use Cases of the *awareness zone* are listed in Figure 68. These use cases do not have a direct safety impact and the maximum latency between the occurrence of an event and the point in time the information has to be processed at the receiver side has a larger range of several seconds. This period can be described in most cases as the time the ego vehicle needs until passing the respective location. However, due to the nature of these use cases, the driver has to be informed up to 30 seconds in advance before passing the respective location. In general, for these use cases the respective application warns or informs the driver about the current situation. For use case realizations multi-hop mobile ad-hoc communication or cellular communication can be the appropriate technology to ensure the necessary communication range. The *awareness zone* has four main subcategories, namely moving objects, cooperative ma-

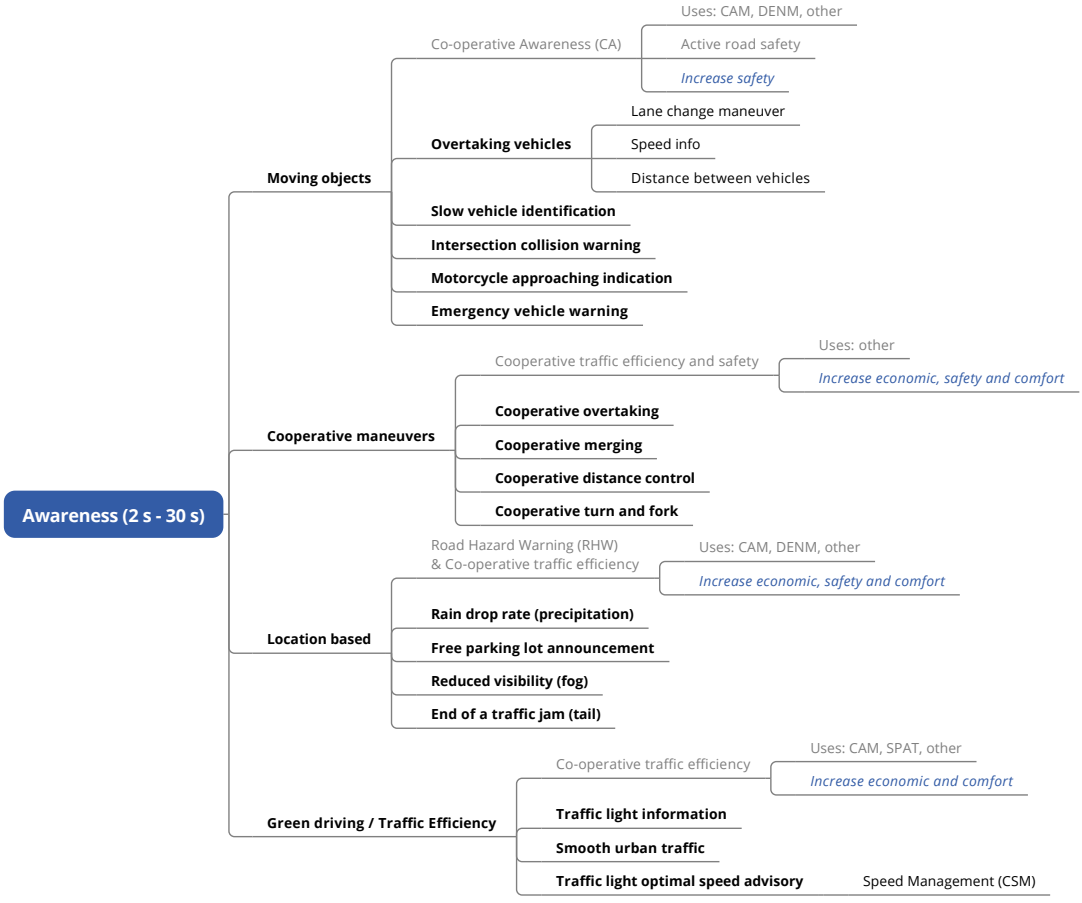

Figure 68: Itemization of the awareness zone related use cases.

neuvers, location based and green driving / traffic efficiency. The respective use case examples are explained in the following.

Moving objects
The main purpose of this subcategory is to increase driving safety. Focus is here the awareness of other moving objects, i.e. road users. In ETSI documents it is categorized as active road safety. Some of these use cases are covered in the EU by the already specified CAM or DENM.

- **Overtaking vehicles:** A vehicle gets informed about overtaking vehicles, which are not in the direct proximity. Thus, the vehicle is approaching a potentially dangerous situation. A slightly slow down of the vehicle can mitigate the situation. The risk of a frontal collision should be reduced. Already in a longer distance the ego vehicle gets informed about speed information of other vehicles, the distance between other vehicles and possible lane change maneuvers. This enables anticipatory driving, e. g., speed reduction of a vehicle on the left lane on an autobahn in case of low distance between vehicles ahead with potential lane change intention, which results in a reduction of collision risk.

- **Slow vehicle identification:** A slow vehicle broadcasts its present position or is detected by another ITS station, i. e., a vehicle or a sensor equipped RSU, that broadcasts the presence. The potential safety risk is mitigated by the awareness of other road users. Informed road users might also be stimulated to use another route that can contribute to an overall increase of traffic flow.

- **Intersection collision warning:** A vehicle approaches an intersection. Information about other vehicles approaching the same intersection from different directions allows to assess the situation. A potential collision risk should be reduced.

- **Motorcycle approaching indication:** A motorcycle broadcasts its presence and status to increase the awareness of other road users. Thus, drivers of other vehicles can be warned of an approaching motorcycle. This is in particular important if visibility is reduced, e. g., by an obstacle. If no line of sight between the vehicles is possible, e. g., obstructed by buildings, a RSU should be installed to relay messages [68]. A potential collision should be avoided.

- **Emergency vehicle warning:** An emergency vehicle periodically broadcasts its presence and status. Informed vehicles can early start to clear an emergency corridor to give way to the approaching emergency vehicle. The intervention time of emergency response should be minimized to protect people. Moreover, the collision risk between emergency vehicles and other road users should be reduced.

Cooperative maneuvers
The main purpose of this subcategory is to increase economic driving, safety and comfort. Focus is here the agreement between road users and the cooperative execution of maneuvers. Until now, there are no message types specified for the realization of these use cases.

- **Cooperative overtaking:** This use case has its main focus on highways and incorporates cooperative overtaking warning and collision prevention. Failed overtaking maneuvers or evasion maneuvers, e. g., because of suddenly appearing deer, on highways often result in severe collisions with oncoming traffic. A correct perception of the situation and estimation of other road users behavior is often difficult. If an overtaking maneuver becomes a dangerous situation, it is difficult to select the correct avoidance strategy. This is because of the dependency of the mutual behavior of all affected road users. Thus, such a system should prevent the emergence of a critical situation. If nevertheless a critical situation happens, the system should prevent an accident.

  Vehicles broadcast position beacons and also a list of objects they perceive with their sensors. Thus, by the use of the received information, the ego vehicle has a better perception and the system can generate a warning to the driver, e. g., if the ego vehicle approaches an overtaking maneuver in the oncoming traffic. If a warning is not possible, e. g., because an overtaking vehicle is not equipped with communication capabilities and not sensed by another communication equipped vehicle, the collision should be prevented by a cooperative maneuver. Such maneuver is calculated cooperatively by the use of communication. All involved vehicles contribute to the prevention of accidents by an intervention in the lateral and longitudinal dynamics of the vehicle.

  A special sub use case is cooperative overtaking of trucks on an autobahn. An exchange of additional information, like power to weight ratio or special speed limits, can help to decide about termination of an overtaking maneuver if conditions change, e. g., a change in road grade.

- **Cooperative merging:** This use case is relevant on driving in and out at ramps in case of lane subtraction or the merge of two lanes. Driving on, e. g., at ramps, is demanding because they occur immediately with no flexibly on a local level. A vehicle has to search for a gap in the flowing traffic. The driver has to simultaneously adapt to the velocity of flowing traffic, with taking preceding vehicles and an approaching end of lane into account. Reachability and prediction of dynamic behavior of an according gap can be challenging. By the use of communication a cooperation partner can be negotiated that opens and guarantees a gap for lane change. Moreover, the exchange of local perception knowledge, i. e., position beacons and a list of objects perceived in the surroundings, allows to estimate gaps.

- **Cooperative distance control:** A cooperative distance control or cooperative adaptive cruise control helps to reduce traffic jams and fuel consumption. Here, a long range cooperation is necessary to ensure a stable traffic. Cooperation based on communication is absolutely necessary. The single use of local sensors is not sufficient to prevent concertina effect caused traffic jams. The system controls longitudinal acceleration and deceleration to ensure a defined distance to the preceding vehicle. The controller not only needs local sensor information, e. g., distance to the preceding vehicle, but also an overall long range information about other road users to prevent concertina effects. The effect of accelerating or decelerating vehicles, as well as lane changes, has to be taken into account as early as possible. An important requirement is the exchange of lo-

cal perception knowledge, i. e., position beacons and a list of objects perceived in the surroundings, to enhance the local perception range of vehicles. Such a system allows to drive closer, with less distance between vehicles. Congestion is reduced, which in turn reduces fuel consumption and emissions.

- **Cooperative turn and fork:** This use case has its main focus at intersections on highways. In particular left turns and driving on a highway are dangerous situations. Here, the driver has to observe several traffic flows with relative high velocity in parallel. In case of low traffic density, the relative velocity is very high. In case of high traffic density, the relative speed decreases, but the amount of potential gaps also decreases. Bad visibility, e. g., obstructed by objects, plants or buildings, can increase difficulty. By the use of communication technology, local perception knowledge, i. e., position beacons and a list of objects perceived in the surroundings, can be exchanged to enhance the local perception range and completeness of vehicles. This allows to better rate the overall situation and hence to reduce the risk of an accident. Moreover, a cooperation partner can be negotiated that opens and guarantees a gap for driving on. In case of the intention to turn left, a broadcast message to succeeding vehicles will allow them to proactively decrease velocity and thus increase distance. This reduces the risk of an accident and also helps to reduce fuel consumption and emissions.

Location based
The main purpose of this subcategory is to increase economic driving, safety and comfort. Focus is here the provision of dynamic location based information. In ETSI documents it is mostly categorized as active road safety. Some of these use cases are covered in the EU by the already specified CAM or DENM.

- **Rain drop rate (precipitation):** This use case focuses on a fluctuating rain drop rate, in particular locally narrow heavy rain. The exchange of information about the local rain drop rate allows to adapt predictively. Vehicle sensor information is provided to succeeding vehicles. In case of a very heavy rain drop rate, also a velocity adaption is necessary. This increases comfort and even safety.

- **Free parking lot announcement:** Vehicles can sense the occupation status of parking spots by driving by or if the ego vehicle leaves a parking place. Information is provided to a cloud or driving by vehicles. This enables vehicles to request information about available parking space. This allows to increase comfort and economy, due to reduced search times for available parking space, especially within bigger cities.

- **Reduced visibility (fog):** Locally reduced visibility, e. g., caused by fog, can cause a hazardous situation, in particular of course if the road is not visible, e. g., behind curves or peaks. The exchange of information about the local view range allows to adapt velocity predictively. Vehicle sensor information is provided to succeeding vehicles, which decreases the risk of an accident.

- **End of a traffic jam (tail):** A typical and dangerous situation is a rear-end collision at the tail of a traffic jam. Local perception knowledge, i. e., position beacons and a list of objects perceived in the surroundings, can be exchanged

by the use of communication technology, to enhance the local perception range and completeness of vehicles. This allows the early identification of the tail of a traffic jam. Velocity can be predictively adapted and thus, the risk of a collision can be reduced.

Green driving / traffic efficiency
The main purpose of this subcategory is to increase economic driving and comfort. Focus is here the realization of traffic efficiency, with respect to economy and environment. Some of these use cases are covered in the EU by the already specified CAM or Signal Phase And Timing Message (SPAT).

- **Traffic light information:** Information about traffic light phases can help to adapt velocity to achieve minimal stops at red lights or even achieve a green wave. An information source can be other vehicles passing a traffic light or in addition a RSU at a traffic light. This increases driving comfort and economy.
- **Smooth urban traffic:** The goal is to achieve evenly distributed utilization of the road network and preferably no congestion. The exchange of local perception knowledge, i. e., position beacons and a list of objects perceived in the surroundings, enhances the local perception range and view completeness of vehicles. This allows to predictively react and adapt. Alternative roads can be used, velocity can be early adapted or a lane can be changed early before it will be blocked by a vehicle with the intention to turn. This increases driving comfort and economy.
- **Traffic light optimal speed advisory:** Especially in urban traffic situations the number of stops and respective acceleration can be reduced to save fuel and reduce emissions. The optimal speed advice is typically given by the provider of the traffic light system, i. e., local government. Such a system allows to increase driving comfort and economy, by a reduction of fuel consumption and emissions.

A.1.3 *Information Zone Use Cases*

Use Cases of the *information zone* are listed in Figure 69. These use cases have no safety impact but focus on driving comfort and economy. However, due to the nature of these use cases, the driver has to be informed up to 30 seconds in advance before passing the respective location. In general, for these use cases the respective application informs the driver about the forthcoming situation in a way that no direct action is required. For use case realizations, cellular communication can be the appropriate technology to ensure the necessary communication range, especially in sparse traffic situations. The *information zone* has three main subcategories, namely flow influencing, road segment information and location based services (LBS). The respective use case examples are explained in the following.

Flow influencing
The main purpose of this subcategory is to increase economic driving, safety and comfort. Focus is here the provision of semi-dynamic location based information. Some of these use cases are covered in the EU by the already specified CAM or SPAT.

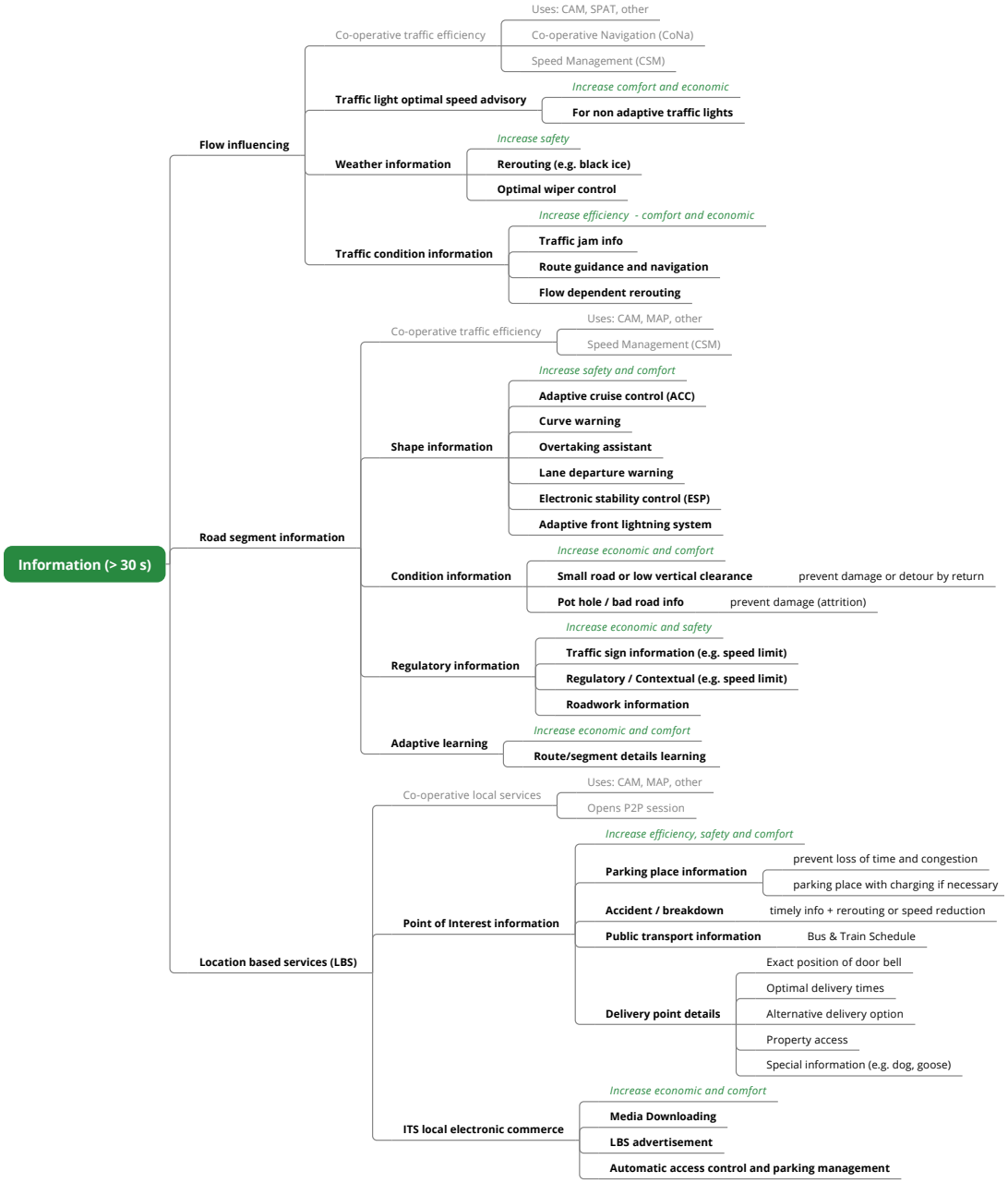

Figure 69: Itemization of the information zone related use cases.

- **Traffic light optimal speed advisory:** This use case focuses on non adaptive traffic lights in situations, where traffic light information is not provided by the respective provider. Vehicles passing a traffic light provide information about traffic light status to the cloud. By aggregation of a multitude of transmitted status information, an algorithm can approximate future traffic light phases and provide respective information as service. Especially in urban traffic situations, the number of stops and respective acceleration can be reduced to save fuel and reduce emissions. Such a system allows to increase driving comfort and economy by a reduction of fuel consumption and emissions.

- **Weather information:** This use case focuses on local weather extremes. Vehicles in local weather extremes, e. g., black ice, provide respective information to the cloud. Based on this information a routing service can suggest a save route around these local weather extremes or provide an according speed warning. This increases comfort and safety.

- **Traffic condition information:** This use case increases efficiency, comfort and economic driving by the provision of traffic condition information. Vehicles or RSUs provide information about current traffic condition at the present position to the cloud. An according service can aggregate received information to provide traffic jam information, route guidance and navigation services and moreover a respective traffic flow dependent rerouting service.

Road segment information
The main purpose of this subcategory is to increase driving safety and comfort. Focus is here the provision of semi-static location based information about road segments. Some of these use cases are partly covered in the EU, by the already specified CAM or MAP.

- **Shape information:** Vehicles sense the shape of road segments they pass and compare against a local map database. In case of deviations, the sensed information is provided to the cloud. Other vehicles can request respective road shape information to increase safety and comfort. Such information is used by several ADASs. An Adaptive Cruise Control (ACC) can adapt velocity to the road shape and a curve warning system can warn the driver if the current velocity is to high for the next curve. An overtaking assistant will need respective information, to suggest an appropriate position to start the overtaking maneuver and a lane departure warning system can adapt to changed road geometries. In extreme cases, an Electronic Stability Control (ESC) can use road shape information to proactively reduce vehicle velocity, if passing the respective road segment would else not be possible. An adaptive front lightning system needs up to date road shape information for optimal light beam control.

- **Condition information:** Vehicles sense the condition of road segments. Bad conditions or low vertical clearance is provided to the cloud. Other vehicles can request respective information, to prevent attrition or even damage. In case of not passable segments, a detour by return can be prevented by early rerouting.

- **Regulatory information:** Vehicles sense regulatory information and provide this to the cloud. This can include traffic signs, e. g., new traffic signs, variable message signs or any other traffic sign. Also regulatory and contextual speed limits or roadwork information. The provision of respective information to other road users and the cloud helps to increase economic driving and safety.

- **Adaptive learning:** This use case allows to adaptively learn road segment information to maintain a map database up to date. Such information is the basis for several assistance systems and helps to increase economic and comfort. Changes in road segments and details about the surroundings can be adaptively learned.

Location based services (LBS)
The main purpose of this subcategory is to increase economic driving and comfort. Focus is here the realization of traffic efficiency, with respect to economy and environment but also to increase driving comfort. Some of these use cases are partly covered in the EU by the already specified CAM or SPAT.

- **Point of Interest information:** This use case provides information about a requested Point of Interest (PoI). An information source can be any road user, RSUs or any other sensor source, e. g., a connected parking garage. The intention is to increase efficiency, safety and comfort. Typically, such information is provided early in a longer distance to enable efficient routing. The early provision of parking place information, including special needs like electric charging facilities, allows to prevent loss of time by detours and helps to prevent congestion caused by vehicles driving around and searching for a free parking place. Information about an accident or a broken-down vehicle allows early rerouting. Public transport information allows efficient multimodal journeys and delivery point details are in particular beneficial for the increasing amount of parcel delivery. Here, the customer, the parcel service and also other road users benefit. Detailed information allows fast and efficient parcel delivery, which in turn relieves the road network.

- **ITS local electronic commerce:** This use case provides location based comfort and entertainment. A RSU broadcasts an advertisement for respective services. The same RSU can also provide media downloading for local electronic commerce or as payed service. In general, a bidirectional communication between vehicles and RSUs brings the capability of local payments for respective services. Such a service could also be the automatic access to charged parking facilities.

A.2 DATA STRUCTURE

In this section, we briefly describe our data structures used within this work. We begin with the serialization structure, designed for data collection, and the provision of the eHorizon in Section A.2.1. Afterwards, we present the data structure of our modified Cooperative Awareness Messages (CAM) in Section A.2.2, followed by the description of our Cooperative Perception Message (CPM) in Section A.2.3.

A.2.1 *Data Collection and eHorizon Serialization Structure*

We have presented our data structure for data collection and the provision of the eHorizon in [147] and [130]. In the following, we give a revised description of our proposed data structure. The structure is implemented in Google Protocol Buffers, because it is very efficient, neutral with respect of the programming language used for other system components, and in particular extendable, i. e., extended structures keep compatible with already implemented code components. The data structure is depicted in Figure 70 as *UML* diagram. In the following, we give a description of this structure, that consists of the five main components *RequestTag*, *Tag*, *RequestInformation*, *Information* and *eHorizon*. All components are optional, i. e., a message does not necessarily consists of all components. The component *RequestTag* is used to specify a request for a specific type of information, typically used to send a request towards mobile nodes, i. e., vehicles. The complementary message component is the *Tag*, typically the response to a request, containing sensed information. The component *Information* is to provide information from the backend towards vehicles. To request this *Information*, vehicles can use the *RequestInformation* component. The fifth component is the *eHorizon*, used to provide an electronic horizon to a vehicle (c.f. Chapter 6). Specific enumerations as types within the single fields are defined in a database, that has to be synchronized between all participants.

Message Header: The message header is the message entry point that relates to the previously mentioned five main components. As mentioned before, all entries are optional. The same holds for some additional fields, the header allows to specify. This way, the header allows to specify an unique message identification (ID), a timestamp, a priority level and a sender identification. The latter one allows to use any unique Id or the vehicle identification number (VIN). Moreover, a message can also contain several objects of a specific type, i. e., an array list.

Request Tag: This component is used to request sensed information from a vehicle. It can contain an unique identifier (ID), that allows to relate the respective response. Moreover, a timestamp, a priority and the type of request, either event or sensor request, can be specified. The component can also contain a list of sensor requests.

Tag: This component is used to transmit sensed information from a vehicle towards the backend. The header of the *Tag* can contain a timestamp, a priority and the respective data values. The *response ID* field allows to relate to a respective request. If the containing values are describing an event, then this is specified by the according event type. The values can consist of either integer values, floating point values or

even text based values. Again, all fields are optional and only the appropriate ones are used, to ensure a maximum efficiency, because it would be inefficient to transmit an integer value inside a float data field. Other optional fields are the sensor type and the respective unit. Each value can also get an accuracy value assigned. If the value is location dependent, is defined by an according flag. Respective position values, i. e., GPS coordinates, are attached as additional values. An example could be a detected speed limit sign. Then, the event type is *speed sign detected* and attached sensor values specify the speed limit and the position.

Request Information: This component is to request a more complex type of information. It can contain an unique identifier (ID), that allows to relate the respective response. Moreover, a timestamp, a priority and the type of request, as *information type*, can be specified. If the request is related to a certain location or area, this can be specified within an optional *location* object. It might be also useful to add some describing values, e. g., sensor readings, to the information request. An example is the request for an *eHorizon*, that gets attached the current velocity and position (c.f. Chapter 6).

Location: The location parameter can be part of the *Request Information* as well as of the response *Information* and can be used in different ways. To relate to a somewhere specified location, the *location ID* field can be used. To set a reference to map material, respective *map ID* fields can be used. In case the referenced map object is a road segment, the *offset* field allows to specify the position on the respective segment. A single position point can be specified by the fields *latitude*, *longitude* and a direction by the field *heading*, as angle relative to true north. To precise the position, a specific lane and an altitude can be specified. The radius field can be used to specify a simple area of a circle. To specify a more complex shape of an area, polygonal chains can be specified by a recursive concatenation of *Location* child objects. This allows to specify an arbitrary angular shaped area as closed polygonal chain or an area, composed of several areas of a circle. An accuracy value allows to specify the preciseness of a given location.

Information: This component is used to transmit any already processed data between entities. The header of the *Information* can contain a timestamp, a priority and the respective data value objects. The *response ID* field allows to relate to a respective request and also the respective information type can be specified. An information can also be associated with a location, defined by the previously described *Location* object. Respective values can be integer, floating point or text based values with an according accuracy. Other optional fields are the information type and the respective unit. Moreover, *Information* objects can be concatenated, i. e., can contain sub objects of type information. An example could be a traffic sign, with the sub type speed limit and the sub type of 'valid time'.

Electronic Horizon: This component is used to specify an eHorizon. It is intended to describe information about the road network, i. e., road geometry and related information, in driving direction. It consists of a graph, constructed as hierarchical tree, that describes the road network. Each tree node represents a road segment. At each intersection a node splits according to the egresses. Only a subset of a complete

eHorizon is transmitted towards a vehicle, ideally only the MPP. An according map matching can be performed on the vehicle side on the known eHorizon, or in case of a deviation from known road segments, on the backend. Therefore, the eHorizon message component supports a server based eHorizon provider, as presented in [147] and described in more detail in Chapter 6. The header consists of an identifier, a type and whether the MPP was determined by local vehicle knowledge or on the backend. General additional information, true for the whole eHorizon, can be attached by a list of *Information* objects. The road network description itself is attached as list of *TreeNodes*. Each *TreeNode* has an unique ID and can be related to a parent node. Thus, the receiver can reconstruct the tree. Related information of a *TreeNode* is attached as a list of *Information* objects. These also describe road geometry by nested location objects. Another option to describe the road geometry is the use of B-Splines, that can be specified by the B-Spline degree and the respective knot vector. Necessary control points are attached via nested *Location* objects inside the respective *Information* object. Details about the construction of B-Splines are given in Chapter 2.6.2. Since the MPP is only a prediction, each possible path in the eHorizon, i. e., *TreeNode*, has given a probability value and it is defined if it belongs to the MPP.

A.2.2 *Extended CAM*

Our CA message, used within Chapter 7, is a modification of the ETSI standardized CAM [35]. The complete ASN.1 definition of this CAM can be found in [35]. In our CAM version we have reused the ETSI standardized CAM ASN.1 definition and extended the high frequency container as depicted in the following listing.

```
...
    BasicVehicleContainerHighFrequency ::= SEQUENCE {
     heading Heading,
     speed Speed,
     driveDirection DriveDirection,
     longitudinalAcceleration LongitudinalAcceleration,
     curvature Curvature,
     curvatureCalculationMode CurvatureCalculationMode,
     yawRate YawRate,
     vehicleLength VehicleLength,
     vehicleWidth VehicleWidth,
     performanceClass PerformanceClass OPTIONAL,
     accelerationControl AccelerationControl OPTIONAL,
     laneNumber LaneNumber OPTIONAL,
     steeringWheelAngle SteeringWheelAngle OPTIONAL,
     lateralAcceleration LateralAcceleration OPTIONAL,
     verticalAcceleration VerticalAcceleration OPTIONAL,
     motionModel MotionModel OPTIONAL,
     laneID UTF8String OPTIONAL
    }
...
```

A.2.3 *Cooperative Perception Message (CPM)*

Our Cooperative Perception Message (CPM), also given as ASN.1 definition, is depicted in the following listing. It imports our modified CAM definition, as introduced in Section A.2.2. Our CPM basically consists of a list of CAM objects, each entry is supplemented with a time stamp of the respective CAM generation time and a lane offset on the underlying map material.

```
CPM-PDU-Descriptions

DEFINITIONS AUTOMATIC TAGS ::= BEGIN

IMPORTS
   CAM, GenerationDeltaTime, ItsPduHeader FROM CAM-PDU-Descriptions {
 itu-t (0) identified-organization (4) etsi (0) itsDomain (5) wg1 (1)
     en (302637) cam (2) version (1)
};

--      The root data frame for cooperative perception messages
CPM ::= SEQUENCE {
        header                  ItsPduHeader,
        cpmList                 CPMList,
        generationDeltaTime     GenerationDeltaTime
}

CPMList ::= SEQUENCE OF CPMListObject

CPMListObject ::= SEQUENCE {
        originalTimestamp       GenerationDeltaTime,
        cam                     CAM,
        positionOnLane          INTEGER(0..4194303) --22 bits ... in cm
             accuracy
}

END
```

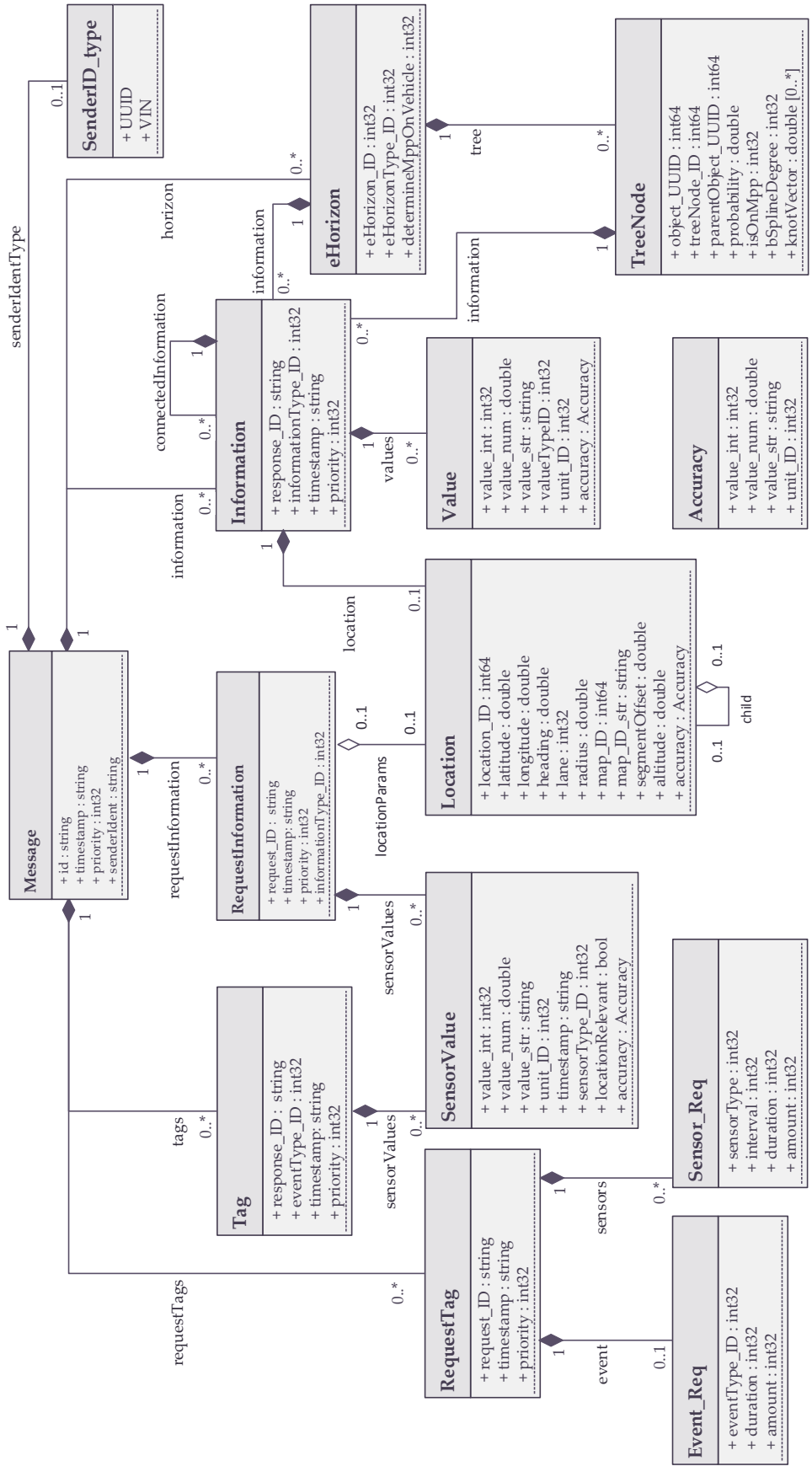

Figure 70: *UML* diagram of our proposed data structure for vehicular information collection and eHorizon provision, as presented in [130].

A.3 EVALUATION RESULTS

In this section we give the evaluation result data, used to plot the figures in Chapter 5 to 7. We start with the evaluation results according to gathering vehicular sensed date in Section A.3.1. Afterwards we present the evaluation results of the provision of the eHorizon as cloud service in Section A.3.2. Finally, we present the evaluation results of our extended vehicular perception mechanism in Section A.3.3.

A.3.1 *Gathering Vehicular Sensed Data*

In the following, we give the evaluation result data of our probabilistic data acquisition approach, presented in Chapter 5.

We start with the result values of *ProbSense.KOM*, as depicted in Figure 11 in Section 5.3.2. The respective events have been placed on a subset of the whole road network of the used scenario. We dropped the road segments with the lowest 5% traffic density. In Table 16, we give the evaluation result data of the comparison of *ProbSense.KOM* with the opportunistic transmission model for discrete events. The Table gives the results for the scenario of Cologne, with a tolerated error of $\alpha = 5\%$ and a detection latency of 10 minutes. In the following Table 17, we give the evaluation result data of the comparison of *ProbSense.KOM* with the opportunistic transmission model for discrete events as depicted in Figure 12 in Section 5.3.2. In this example, we also dropped for event placement the road segments with the lowest 5% traffic density. The Table gives the results for the scenario of Cologne, with a tolerated error of $\alpha = 1\%$ and a detection latency of 10 minutes.

Next, we consider *ProbSense.KOM*, but events are placed randomly throughout the whole scenario, as depicted in Figure 13. In Table 18, we give the evaluation result data of the comparison of *ProbSense.KOM* with the opportunistic transmission model for discrete events. The Table gives the results for the scenario of Cologne, with a tolerated error of $\alpha = 5\%$ and a detection latency of 10 minutes. In the following Table 19, we give the evaluation result data of the comparison of *ProbSense.KOM* with the opportunistic transmission model for discrete events as depicted in Figure 14 in Section 5.3.2. In this example events are completely randomly placed throughout the scenario. The Table gives the results for the scenario of Cologne, with a tolerated error of $\alpha = 1\%$ and a detection latency of 10 minutes.

Afterwards, we consider different tolerated detection latencies, as depicted in Figure 15 in Section 5.3.2. In Table 20, we give the evaluation result data of the comparison of *ProbSense.KOM* with the opportunistic transmission model for discrete events. The Table gives the results for the scenario of Cologne, with a tolerated error of $\alpha = 5\%$ and $\alpha = 1\%$, for a detection latency of 5, 10 and 20 minutes and 8x8 geo cells. In this example, we dropped the road segments with the lowest 5% traffic density for event placement.

HYBRID-PROBSENSE.KOM: Next, we give the evaluation results of our data gathering approach based on hybrid communication, i. e., *Hybrid-ProbSense.KOM*, as depicted in Figure 16 in Section 5.3.2. In Table 21 we give a comparison of different V2V penetration rates in *Hybrid-ProbSense.KOM* with the opportunistic transmission model for discrete events. The Table gives the results for the scenario of Cologne,

with a tolerated error of $\alpha = 5\%$, for a detection latency 10 minutes and 8x8 geo cells. In this example, we dropped the road segments with the lowest 5% traffic density for event placement.

Afterwards, we give a comparison of the opportunistic transmission model, *ProbSense.KOM* and *Hybrid-ProbSense.KOM* at different traffic densities, as depicted in Figure 17. In Table 22 we give the results for the scenario of Cologne, with a tolerated error of $\alpha = 5\%$, for a desired detection latency of 10 minutes and 8x8 geo cells. In this example, we dropped the road segments with the lowest 5% traffic density for event placement.

CONTINUOUS EVENTS: Next, we consider *ProbSense.KOM* for continuous events, as depicted in Figure 19 in Section 5.3.2. In Table 23 we give a comparison of *ProbSense.KOM* with the opportunistic transmission model for continuous events. The Table gives the results for the scenario of Cologne for continuous events, with a sensing density of 20 $\frac{meas.}{km^2 \cdot h}$. In this example, we dropped the road segments with the lowest 5% traffic density for event placement.

In the following, we give a comparison of *ProbSense.KOM* with the opportunistic transmission model for continuous events for different sensing densities of 5, 10 and 20 $[\frac{meas.}{km^2 \cdot h}]$, as depicted in Figure 20 in Section 5.3.2. In Table 24 we give a comparison of *ProbSense.KOM*, in the setting of 8x8 geo cells, for the scenario of Cologne for continuous events for different sensing densities of 5, 10 and 20 $[\frac{meas.}{km^2 \cdot h}]$. In this example, we dropped the road segments with the lowest 5% traffic density for event placement.

LUXEMBOURG SCENARIO: Finally we give a comparison of *ProbSense.KOM* and *Hybrid-ProbSense.KOM* with the opportunistic transmission model for discrete events, as depicted in Figure 21 in Section 5.3.2. The Table Table 25 gives the results for the scenario of Luxembourg for discrete events with a tolerated error of $\alpha = 5\%$ and a detection latency of 10 minutes. The V2V penetration rate in case of *Hybrid-ProbSense.KOM* is set to 100%. In this example events are completely randomly placed throughout the scenario.

Table 16: Comparison of *ProbSense.KOM* with the opportunistic transmission model for discrete events. The table gives the results for the scenario of Cologne, with a tolerated error of $\alpha = 5\%$ and a detection latency of 10 minutes. In this example, we dropped the road segments with the lowest 5% traffic density for event placement.

Run configuration	**Event traffic**	**Lower confidence value**	**Upper confidence value**	**Control traffic**	**Lower confidence value**	**Upper confidence value**
Non Optimized	97.753 MB	97.016 MB	98.489 MB	0.000 MB	0.000 MB	0.000 MB
1x1 geo cells	20.860 MB	18.854 MB	22.865 MB	2.120 MB	1.984 MB	2.257 MB
2x2 geo cells	35.188 MB	31.552 MB	38.823 MB	1.990 MB	1.886 MB	2.095 MB
4x4 geo cells	41.225 MB	38.190 MB	44.259 MB	1.784 MB	1.695 MB	1.873 MB
8x8 geo cells	59.822 MB	56.410 MB	63.235 MB	1.198 MB	1.126 MB	1.270 MB
16x16 geo cells	80.535 MB	77.535 MB	83.534 MB	0.480 MB	0.422 MB	0.538 MB

Table 17: Comparison of *ProbSense.KOM* with the opportunistic transmission model for discrete events. The table gives the results for the scenario of Cologne, with a tolerated error of $\alpha = 1\%$ and a detection latency of 10 minutes. In this example, we dropped the road segments with the lowest 5% traffic density for event placement.

Run configuration	**Event traffic**	**Lower confidence value**	**Upper confidence value**	**Control traffic**	**Lower confidence value**	**Upper confidence value**
Non Optimized	97.753 MB	97.016 MB	98.489 MB	0.000 MB	0.000 MB	0.000 MB
1x1 geo cells	21.495 MB	18.542 MB	24.448 MB	2.299 MB	2.239 MB	2.358 MB
2x2 geo cells	35.422 MB	32.469 MB	38.375 MB	1.993 MB	1.933 MB	2.052 MB
4x4 geo cells	41.707 MB	38.754 MB	44.660 MB	1.807 MB	1.747 MB	1.866 MB
8x8 geo cells	60.119 MB	56.474 MB	63.764 MB	1.226 MB	1.166 MB	1.285 MB
16x16 geo cells	80.301 MB	77.349 MB	83.254 MB	0.484 MB	0.424 MB	0.543 MB

Table 18: Comparison of *ProbSense.KOM* with the opportunistic transmission model for discrete events. The table gives the results for the scenario of Cologne, with a tolerated error of $\alpha = 5\%$ and a detection latency of 10 minutes. In this example events are completely randomly placed throughout the scenario.

Run configuration	Event traffic	Lower confidence value	Upper confidence value	Control traffic	Lower confidence value	Upper confidence value
Non Optimized	347.123 MB	346.130 MB	348.116 MB	0.000 MB	0.000 MB	0.000 MB
1x1 geo cells	347.123 MB	346.130 MB	348.116 MB	0.000 MB	0.000 MB	0.000 MB
2x2 geo cells	326.345 MB	322.746 MB	329.944 MB	2.110 MB	1.811 MB	2.409 MB
4x4 geo cells	317.786 MB	314.479 MB	321.093 MB	3.209 MB	2.996 MB	3.421 MB
8x8 geo cells	256.333 MB	250.664 MB	262.002 MB	6.606 MB	6.411 MB	6.801 MB
16x16 geo cells	258.585 MB	253.987 MB	263.183 MB	7.875 MB	7.609 MB	8.140 MB
32x32 geo cells	305.698 MB	302.969 MB	308.427 MB	4.138 MB	3.901 MB	4.374 MB

Table 19: Comparison of *ProbSense.KOM* with the opportunistic transmission model for discrete events. The table gives the results for the scenario of Cologne, with a tolerated error of $\alpha = 1\%$ and a detection latency of 10 minutes. In this example events are completely randomly placed throughout the scenario.

Run configuration	Event traffic	Lower confidence value	Upper confidence value	Control traffic	Lower confidence value	Upper confidence value
Non Optimized	347.123 MB	346.130 MB	348.116 MB	0.000 MB	0.000 MB	0.000 MB
1x1 geo cells	347.123 MB	346.130 MB	348.116 MB	0.000 MB	0.000 MB	0.000 MB
2x2 geo cells	332.425 MB	329.970 MB	334.880 MB	1.944 MB	1.693 MB	2.194 MB
4x4 geo cells	328.424 MB	324.920 MB	331.927 MB	2.642 MB	2.341 MB	2.943 MB
8x8 geo cells	267.092 MB	261.225 MB	272.959 MB	6.499 MB	6.224 MB	6.774 MB
16x16 geo cells	264.715 MB	260.133 MB	269.298 MB	7.864 MB	7.595 MB	8.133 MB
32x32 geo cells	308.562 MB	305.453 MB	311.671 MB	4.075 MB	3.854 MB	4.296 MB

Table 20: Comparison of *ProbSense.KOM* with the opportunistic transmission model for discrete events. The table gives the results for the scenario of Cologne, with a tolerated error of $\alpha = 5\%$ and $\alpha = 1\%$, for a detection latency of 5, 10 and 20 minutes and 8x8 geo cells. In this example, we dropped the road segments with the lowest 5% traffic density for event placement.

Run configuration	**Event traffic**	**Lower confidence value**	**Upper confidence value**	**Control traffic**	**Lower confidence value**	**Upper confidence value**
Non Optimized MB	97,753 MB	97,016 MB	98,489 MB	0,000 MB	0,000 MB	0,000 MB
5 min. detection latency, $\alpha = 5\%$	52,159 MB	47,718 MB	56,601 MB	1,735 MB	1,601 MB	1,870 MB
10 min. detection latency, $\alpha = 5\%$	59,822 MB	56,410 MB	63,235 MB	1,198 MB	1,126 MB	1,270 MB
20 min. detection latency, $\alpha = 5\%$	79,080 MB	76,409 MB	81,750 MB	0,502 MB	0,438 MB	0,567 MB
5 min. detection latency, $\alpha = 1\%$	55,093 MB	50,753 MB	59,432 MB	1,697 MB	1,571 MB	1,824 MB
10 min. detection latency, $\alpha = 1\%$	60,119 MB	56,474 MB	63,764 MB	1,226 MB	1,166 MB	1,285 MB
20 min. detection latency, $\alpha = 1\%$	78,898 MB	75,884 MB	81,912 MB	0,504 MB	0,504 MB	0,504 MB

Table 21: Comparison of different V2V penetration rates in *Hybrid-ProbSense.KOM* with the opportunistic transmission model for discrete events. The table gives the results for the scenario of Cologne, with a tolerated error of $\alpha = 5\%$, for a detection latency 10 minutes and 8x8 geo cells. In this example, we dropped the road segments with the lowest 5% traffic density for event placement.

Run configuration	**Event traffic**	**Lower confidence value**	**Upper confidence value**	**Control traffic**	**Lower confidence value**	**Upper confidence value**
Non Optimized	97.753 MB	97.016 MB	98.489 MB	0.000 MB	0.000 MB	0.000 MB
V2V rate = 0 %	59.822 MB	56.410 MB	63.235 MB	1.198 MB	1.126 MB	1.270 MB
V2V rate = 25 %	57.293 MB	53.889 MB	60.696 MB	1.188 MB	1.121 MB	1.256 MB
V2V rate = 50 %	54.801 MB	51.270 MB	58.332 MB	1.195 MB	1.122 MB	1.269 MB
V2V rate = 75 %	53.952 MB	50.601 MB	57.302 MB	1.097 MB	1.040 MB	1.155 MB
V2V rate= 100 %	53.979 MB	50.524 MB	57.433 MB	1.100 MB	1.043 MB	1.156 MB

Table 22: Comparison of the opportunistic transmission model, *ProbSense.KOM* and *Hybrid-ProbSense.KOM* at different traffic densities. The table gives the results for the scenario of Cologne, with a tolerated error of $\alpha = 5\%$, for a desired detection latency of 10 minutes and 8x8 geo cells. In this example, we dropped the road segments with the lowest 5% traffic density for event placement.

Run configuration	**Event traffic**	**Control traffic**
Non Optimized, low traffic density	0.406 MB	0.000 MB
ProbSense.KOM, low traffic density	0.406 MB	0.000 MB
Hybrid-ProbSense.KOM, low traffic density	0.406 MB	0.010 MB
Non Optimized, mid traffic density	7.387 MB	0.000 MB
ProbSense.KOM, mid traffic density	4.571 MB	0.299 MB
Hybrid-ProbSense.KOM, mid traffic density	2.120 MB	0.373 MB
Non Optimized, high traffic density	30.743 MB	0.000 MB
ProbSense.KOM, high traffic density	13.975 MB	0.839 MB
Hybrid-ProbSense.KOM, high traffic density	5.573 MB	0.779 MB

Table 23: Comparison of *ProbSense.KOM* with the opportunistic transmission model for continuous events. The table gives the results for the scenario of Cologne for continuous events, with a sensing density of 20 $\frac{\mathrm{meas.}}{\mathrm{km}^2 \cdot \mathrm{h}}$. In this example, we dropped the road segments with the lowest 5% traffic density for event placement.

Run configuration	**Event traffic**	**Lower confidence value**	**Upper confidence value**	**Control traffic**	**Lower confidence value**	**Upper confidence value**
Non Optimized	674.131 MB	671.236 MB	677.025 MB	0.000 MB	0.000 MB	0.000 MB
1x1 geo cells	15.528 MB	15.467 MB	15.589 MB	8.903 MB	8.807 MB	8.999 MB
2x2 geo cells	15.380 MB	15.349 MB	15.411 MB	12.525 MB	12.442 MB	12.608 MB
4x4 geo cells	12.751 MB	12.736 MB	12.766 MB	16.387 MB	16.245 MB	16.530 MB
8x8 geo cells	11.191 MB	11.175 MB	11.206 MB	23.175 MB	23.060 MB	23.289 MB
16x16 geo cells	21.688 MB	20.799 MB	22.578 MB	34.308 MB	34.199 MB	34.418 MB

Table 24: Comparison of *ProbSense.KOM* with the opportunistic transmission model for continuous events. The table gives the results for the scenario of Cologne for continuous events for different sensing densities of 5, 10 and 20 $[\frac{meas.}{km^2 \cdot h}]$, in the setting of 8x8 geo cells. In this example, we dropped the road segments with the lowest 5% traffic density for event placement.

Run configuration	**Event traffic**	**Lower confidence value**	**Upper confidence value**	**Control traffic**	**Lower confidence value**	**Upper confidence value**
Non Optimized	674,131 MB	671,236 MB	677,025 MB	0,000 MB	0,000 MB	0,000 MB
5 $\frac{meas.}{km^2 \cdot h}$	8,685 MB	8,673 MB	8,697 MB	0,171 MB	8,907 MB	9,249 MB
10 $\frac{meas.}{km^2 \cdot h}$	11,269 MB	11,238 MB	11,300 MB	0,131 MB	8,876 MB	9,138 MB
20 $\frac{meas.}{km^2 \cdot h}$	15,528 MB	15,467 MB	15,589 MB	0,096 MB	8,807 MB	8,999 MB

Table 25: Comparison of *ProbSense.KOM* and *Hybrid-ProbSense.KOM* with the opportunistic transmission model for discrete events. The Table gives the results for the scenario of Luxembourg for discrete events with a tolerated error of $\alpha = 5\%$ and a detection latency of 10 minutes. The V2V penetration rate in case of *Hybrid-ProbSense.KOM* is set to 100%. In this example events are completely randomly placed throughout the scenario.

Run configuration	**Event traffic**	**Control traffic**
Non Optimized, 8x8 geo cells	2551.960 MB	0.000 MB
ProbSense.KOM, 8x8 geo cells	1837.390 MB	44.844 MB
Hybrid-ProbSense.KOM, 8x8 geo cells, V2V rate = 100%	607.605 MB	76.680 MB

A.3.2 *The Provision of an eHorizon as Cloud Service*

In the following we give the evaluation result data of our eHorizon mechanism, presented in Chapter 6. We start with the empirical determination of the weighting parameters α and β of our MPP heuristic, as visualized in Figure 28 in Section 6.3. In Table 26 we give the evaluation result data of the correctness of our MPP heuristic in the scenario of *TAPAS Cologne* for different values of α and β. In the following, we analyze the average amount of necessary messages for the eHorizon transmission of the selected set of traces, as visualized in Figure 29. In Table 27 we give the evaluation result data of the average amount of messages in comparison of a completely known MPP and the usage of our MPP heuristic, with $\alpha = 0.9$ and $\beta = 0.1$. Each for an eHorizon depth of $0, 1, 2$, and 3 and the three selected groups of traces.

Table 26: Correctness of our MPP heuristic in the scenario of *TAPAS Cologne* for different values of α and β.

Run configuration	**Mean**	**Lower confidence value**	**Upper confidence value**
$\alpha = 0.0; \beta = 1.0$	0.559 %	0.558 %	0.560 %
$\alpha = 0.1; \beta = 0.9$	0.559 %	0.558 %	0.560 %
$\alpha = 0.2; \beta = 0.8$	0.559 %	0.558 %	0.560 %
$\alpha = 0.3; \beta = 0.7$	0.561 %	0.560 %	0.562 %
$\alpha = 0.4; \beta = 0.6$	0.565 %	0.564 %	0.566 %
$\alpha = 0.5; \beta = 0.5$	0.584 %	0.583 %	0.584 %
$\alpha = 0.6; \beta = 0.4$	0.597 %	0.596 %	0.598 %
$\alpha = 0.7; \beta = 0.3$	0.611 %	0.610 %	0.612 %
$\alpha = 0.8; \beta = 0.2$	0.627 %	0.626 %	0.628 %
$\alpha = 0.9; \beta = 0.1$	0.628 %	0.627 %	0.629 %
$\alpha = 1.0; \beta = 0.0$	0.329 %	0.328 %	0.330 %

Table 27: Average amount of messages in comparison of a completely known MPP and the usage of our MPP heuristic with $\alpha = 0.9$ and $\beta = 0.1$. Each for an eHorizon depth of $0, 1, 2$, and 3 and the three selected groups of traces.

Run configuration	Mean MPP known	Lower confidence value	Upper confidence value	Mean MPP unknown	Lower confidence value	Upper confidence value
short D=0	187.592	167.745	207.439	201.840	181.088	222.592
short D=1	170.585	150.779	190.391	185.840	165.626	206.054
short D=2	152.940	137.212	168.668	155.810	140.188	171.432
short D=3	123.140	110.783	135.497	130.345	116.290	144.400
middle D=0	364.805	334.830	394.780	404.500	340.850	468.150
middle D=1	333.909	296.451	371.367	383.000	345.542	420.458
middle D=2	319.167	282.504	355.829	337.111	294.541	379.682
middle D=3	286.091	252.181	320.001	292.045	256.057	328.034
long D=0	428.565	394.030	463.100	442.540	409.241	475.839
long D=1	412.545	386.821	438.269	437.568	404.269	470.867
long D=2	397.263	376.210	418.316	411.951	381.801	442.101
long D=3	361.733	344.445	379.021	373.254	348.391	398.117

A.3.3 *Extended Vehicular Perception*

In the following, we give the evaluation result data of our extended vehicular perception mechanism, presented in Chapter 7. All tables show the comparison of standardized ETSI ITS G5 CAM implementation to our approach for all different evaluation configurations, as described in Section 7.3.1.3 in Table 14.

NUMBER OF GENERATED CAMS: We start with the result values of the average number of generated CAM messages, visualized in Figure 36 in Section 7.3.2.1. In Table 28 we give the evaluation result of the average number of generated CA messages in a scenario with 250 vehicles, followed by Table 29 with the respective results in a scenario with 500 vehicles. In Table 30 we give the evaluation result of the average number of generated CAM messages in a scenario with 1000 vehicles, followed by Table 31 with the respective results in a scenario with 1500 vehicles.

AVERAGE CBR: In the following, we give the evaluation result values of the average Channel Busy Ratio (CBR), visualized in Figure 41 in Section 7.3.2.1. In Table 32 we give the evaluation result of the average Channel Busy Ratio in a scenario with 250 vehicles, followed by Table 33 with the respective results in a scenario with 500 vehicles. In Table 34 we give the evaluation result of the average Channel Busy Ratio in a scenario with 1000 vehicles, followed by Table 35 with the respective results in a scenario with 1500 vehicles.

AVERAGE PER: Afterwards, we give in Table 36 the evaluation result values of the average Packet Error Rate (PER) as comparison of standardized ETSI ITS G5 CAM implementation to our approach, visualized in Figure 42 in Section 7.3.2.1.

TRIGGER CONDITIONS: Next, we give the evaluation result values of the average trigger condition to send a new CAM, visualized in Figures 43, 44, 45 and 46 in

Section 7.3.2.1. As described in Table 15 in Section 7.3.2.1, condition 1 is triggered by a change in the used kinematic model, condition 2 is triggered if the time that has passed since the last broadcast of a CAM has exceeded the minimal CAM sending rate, condition 3 is triggered by a lane or road change and condition 4 is triggered if the error in prediction exceeds the threshold ε. In Table 37 we give the evaluation result of the average mean for trigger conditions to send a new CAM in a scenario with 250 vehicles, followed by Table 38 with the respective results in a scenario with 500 vehicles. In Table 39 we give the evaluation result of the average mean for trigger conditions to send a new CAM in a scenario with 1000 vehicles, followed by Table 40 with the respective results in a scenario with 1500 vehicles.

PERCEPTION RANGE: After giving the evaluation result values of our CAM mechanism, we continue with the evaluation results of a perception extension mechanism, i. e., broadcasting CPMs. We start in Table 41 with the results of the average of the maximum perception range, visualized in Figure 47.

COMPLETENESS: Next, we give the evaluation result values of the average completeness of local vehicle knowledge, with respect to the ratio of known vehicles to existing vehicles in the respective distance, visualized in Figures 48, 49, 50 and 51 in Section 7.3.2.2. In Table 42, we give the evaluation result of the average completeness of local vehicle knowledge in a scenario with 250 vehicles, followed by Table 43 with the respective results in a scenario with 500 vehicles. In Table 44, we give the evaluation result of the average completeness of local vehicle knowledge in a scenario with 1000 vehicles, followed by Table 45 with the respective results in a scenario with 1500 vehicles.

AVERAGE CBR AND LOCAL VEHICLE DATABASE: In the following, we give this we give in Table 46 the evaluation result values of the average Channel Busy Ratio (CBR) with the use of CPMs, as visualized in Figure 56. Afterwards, we give the evaluation result values of the average amount of known other vehicles in the local vehicle database, with and without the use of CPMs, visualized in Figures 57, 58, 59 and 60 in Section 7.3.2.2. In Table 47, we give the evaluation result of the average amount of known other vehicles in the local vehicle database in a scenario with 250 vehicles, followed by Table 48 with the respective results in a scenario with 500 vehicles. In Table 49, we give the evaluation result of the average amount of known other vehicles in the local vehicle database in a scenario with 1000 vehicles, followed by Table 50 with the respective results in a scenario with 1500 vehicles.

AVERAGE PER: Afterwards, we give in Table 51 the evaluation results values of the average Packet Error Rate (PER) of our approach, with the use of CPMs, as visualized in Figure 61. The figure gives the results for a trigger check frequency of 100ms and all four vehicles densities of 250, 500, 1000 and 1500 vehicles.

INFLUENCE OF TRIGGER CHECK FREQUENCY: Finally, we give the evaluation result values of our analysis of the influence of a reduced trigger check interval, reduced from 100ms to 10ms. In Table 52, we give evaluation result values of the respective comparison of the average number of generated CAM messages for a scenario with 500 vehicles, as visualized in Figure 62. Next, we give in Table 53 the

evaluation result values of the average CBR with a trigger check interval of 100ms and 10ms, as visualized in Figure 63. Following this, we give in Table 54 the evaluation results values of the average PER with a trigger check interval of 100ms and 10ms, as visualized in Figure 64.

Table 28: Evaluations results: Average number of generated CAM messages in a scenario with 250 vehicles.

Run configuration	Mean	Lower confidence value	Upper confidence value
r.0	2445.220	2439.650	2450.790
r.1	335.824	335.520	336.128
r.2	340.000	339.566	340.434
r.3	347.504	346.813	348.195
r.4	333.720	333.490	333.950
r.5	337.808	337.449	338.167
r.6	342.448	341.965	342.931
r.7	332.920	332.711	333.129
r.8	337.088	336.751	337.425
r.9	340.272	339.853	340.691
r.10	184.384	183.877	184.891
r.11	190.708	189.945	191.471
r.12	202.392	201.204	203.580
r.13	181.084	180.755	181.413
r.14	186.796	186.218	187.374
r.15	195.164	194.386	195.942
r.16	179.628	179.401	179.855
r.17	185.192	184.723	185.662
r.18	189.652	189.101	190.203

Table 29: Evaluations results: Average number of generated CAM messages in a scenario with 500 vehicles.

Run configuration	Mean	Lower confidence value	Upper confidence value
r.0	2373.532	2369.318	2377.746
r.1	341.548	341.122	341.974
r.2	347.824	347.075	348.573
r.3	348.820	348.201	349.439
r.4	338.606	338.177	339.035
r.5	342.802	342.103	343.501
r.6	339.280	338.753	339.807
r.7	337.440	337.027	337.854
r.8	341.714	341.079	342.349
r.9	336.822	336.314	337.330
r.10	195.798	195.063	196.533
r.11	211.794	210.783	212.805
r.12	216.126	215.553	216.699
r.13	188.584	188.085	189.083
r.14	197.694	196.828	198.560
r.15	196.750	196.175	197.325
r.16	185.572	185.127	186.017
r.17	191.468	190.646	192.290
r.18	186.428	185.788	187.068

Table 30: Evaluations results: Average number of generated CAM messages in a scenario with 1000 vehicles.

Run configuration	Mean	Lower confidence value	Upper confidence value
r.0	1746.143	1739.854	1752.432
r.1	336.722	336.402	337.0424
r.2	334.191	333.967	334.415
r.3	336.734	336.526	336.942
r.4	335.314	335.069	335.559
r.5	331.825	331.700	331.950
r.6	333.507	333.392	333.622
r.7	334.465	334.266	334.664
r.8	331.751	331.634	331.868
r.9	331.908	331.828	331.988
r.10	193.294	192.744	193.844
r.11	196.576	196.143	197.009
r.12	200.48	200.0353	200.925
r.13	181.603	181.256	181.950
r.14	179.792	179.590	179.994
r.15	182.721	182.533	182.909
r.16	180.696	180.391	181.001
r.17	177.455	177.296	177.614
r.18	178.05	177.944	178.156

Table 31: Evaluations results: Average number of generated CAM messages in a scenario with 1500 vehicles.

Run configuration	Mean	Lower confidence value	Upper confidence value
r.0	1113.632	1111.059	1116.205
r.1	329.589	329.534	329.645
r.2	329.543	329.494	329.593
r.3	331.202	331.142	331.262
r.4	329.398	329.349	329.447
r.5	329.275	329.233	329.318
r.6	330.490	330.448	330.532
r.7	329.368	329.320	329.416
r.8	329.267	329.225	329.309
r.9	330.319	330.279	330.359
r.10	176.047	175.949	176.144
r.11	177.118	176.994	177.242
r.12	179.567	179.422	179.713
r.13	174.201	174.140	174.261
r.14	174.529	174.469	174.588
r.15	176.383	176.318	176.447
r.16	174.032	173.974	174.090
r.17	174.042	173.994	174.091
r.18	175.297	175.254	175.341

Table 32: Evaluations results: Average Channel Busy Ratio (CBR) in a scenario with 250 vehicles.

Run configuration	Mean	Lower confidence value	Upper confidence value
r.0	8.272 %	8.140 %	8.404 %
r.1	1.110 %	1.093 %	1.128 %
r.2	1.114 %	1.096 %	1.131 %
r.3	1.134 %	1.115 %	1.152 %
r.4	1.106 %	1.089 %	1.123 %
r.5	1.108 %	1.091 %	1.126 %
r.6	1.123 %	1.105 %	1.141 %
r.7	1.104 %	1.087 %	1.121 %
r.8	1.108 %	1.091 %	1.125 %
r.9	1.119 %	1.101 %	1.136 %
r.10	0.611 %	0.601 %	0.621 %
r.11	0.626 %	0.616 %	0.637 %
r.12	0.651 %	0.640 %	0.662 %
r.13	0.601 %	0.591 %	0.610 %
r.14	0.613 %	0.603 %	0.623 %
r.15	0.628 %	0.618 %	0.639 %
r.16	0.596 %	0.586 %	0.605 %
r.17	0.610 %	0.600 %	0.620 %
r.18	0.615 %	0.605 %	0.625 %

Table 33: Evaluations results: Average Channel Busy Ratio (CBR) in a scenario with 500 vehicles.

Run configuration	Mean	Lower confidence value	Upper confidence value
r.0	15.497 %	15.381 %	15.613 %
r.1	2.174 %	2.156 %	2.193 %
r.2	2.203 %	2.185 %	2.221 %
r.3	2.203 %	2.187 %	2.220 %
r.4	2.166 %	2.147 %	2.184 %
r.5	2.179 %	2.162 %	2.196 %
r.6	2.148 %	2.132 %	2.165 %
r.7	2.157 %	2.139 %	2.175 %
r.8	2.172 %	2.155 %	2.189 %
r.9	2.137 %	2.121 %	2.153 %
r.10	1.258 %	1.245 %	1.270 %
r.11	1.353 %	1.339 %	1.366 %
r.12	1.365 %	1.353 %	1.378 %
r.13	1.212 %	1.200 %	1.223 %
r.14	1.259 %	1.248 %	1.270 %
r.15	1.243 %	1.233 %	1.252 %
r.16	1.197 %	1.186 %	1.208 %
r.17	1.222 %	1.212 %	1.232 %
r.18	1.183 %	1.175 %	1.192 %

Table 34: Evaluations results: Average Channel Busy Ratio (CBR) in a scenario with 1000 vehicles.

Run configuration	Mean	Lower confidence value	Upper confidence value
r.0	21.417 %	21.379 %	21.454 %
r.1	4.160 %	4.152 %	4.167 %
r.2	4.122 %	4.114 %	4.129 %
r.3	4.128 %	4.120 %	4.135 %
r.4	4.146 %	4.139 %	4.154 %
r.5	4.102 %	4.095 %	4.110 %
r.6	4.114 %	4.107 %	4.122 %
r.7	4.137 %	4.129 %	4.145 %
r.8	4.107 %	4.100 %	4.115 %
r.9	4.097 %	4.089 %	4.105 %
r.10	2.466 %	2.462 %	2.469 %
r.11	2.503 %	2.499 %	2.507 %
r.12	2.523 %	2.519 %	2.527 %
r.13	2.327 %	2.324 %	2.331 %
r.14	2.309 %	2.305 %	2.314 %
r.15	2.318 %	2.314 %	2.322 %
r.16	2.317 %	2.313 %	2.320 %
r.17	2.289 %	2.284 %	2.293 %
r.18	2.282 %	2.277 %	2.286 %

Table 35: Evaluations results: Average Channel Busy Ratio (CBR) in a scenario with 1500 vehicles.

Run configuration	Mean	Lower confidence value	Upper confidence value
r.0	21.539 %	21.519 %	21.559 %
r.1	5.827 %	5.823 %	5.832 %
r.2	5.809 %	5.804 %	5.813 %
r.3	5.795 %	5.790 %	5.799 %
r.4	5.801 %	5.796 %	5.805 %
r.5	5.804 %	5.799 %	5.808 %
r.6	5.790 %	5.786 %	5.795 %
r.7	5.812 %	5.807 %	5.816 %
r.8	5.821 %	5.816 %	5.825 %
r.9	5.811 %	5.806 %	5.816 %
r.10	3.335 %	3.332 %	3.338 %
r.11	3.341 %	3.338 %	3.344 %
r.12	3.336 %	3.333 %	3.339 %
r.13	3.323 %	3.320 %	3.325 %
r.14	3.313 %	3.310 %	3.316 %
r.15	3.323 %	3.320 %	3.326 %
r.16	3.319 %	3.317 %	3.322 %
r.17	3.318 %	3.315 %	3.321 %
r.18	3.319 %	3.316 %	3.322 %

Table 36: Evaluations results: Average Packet Error Rate (PER) as comparison of standardized ETSI ITS G5 CAM implementation to our approach for all different evaluation run configurations r.1 to r.18, each for the scenario with 250, 500, 1000 and 1500 vehicles.

Run configuration	Mean 250 vehicles	Lower confidence value	Upper confidence value	Mean 500 vehicles	Lower confidence value	Upper confidence value	Mean 1000 vehicles	Lower confidence value	Upper confidence value	Mean 1500 vehicles	Lower confidence value	Upper confidence value
r.0	5.19 %	5.05 %	5.33 %	9.38 %	9.29 %	9.47 %	12.31 %	12.28 %	12.34 %	12.63 %	12.61 %	12.64 %
r.1	9.48 %	9.23 %	9.73 %	14.69 %	14.53 %	14.86 %	26.23 %	26.17 %	26.29 %	37.93 %	37.89 %	37.97 %
r.2	11.01 %	10.71 %	11.32 %	14.58 %	14.43 %	14.73 %	26.57 %	26.51 %	26.64 %	38.49 %	38.46 %	38.53 %
r.3	10.22 %	9.95 %	10.49 %	14.69 %	14.51 %	14.86 %	26.52 %	26.44 %	26.59 %	38.74 %	38.70 %	38.78 %
r.4	9.54 %	9.27 %	9.81 %	14.75 %	14.59 %	14.91 %	26.78 %	26.72 %	26.84 %	39.02 %	38.98 %	39.06 %
r.5	10.86 %	10.57 %	11.15 %	14.63 %	14.47 %	14.78 %	27.02 %	26.94 %	27.10 %	38.69 %	38.65 %	38.72 %
r.6	10.14 %	9.87 %	10.41 %	14.91 %	14.73 %	15.10 %	26.13 %	26.05 %	26.21 %	38.87 %	38.84 %	38.91 %
r.7	10.09 %	9.82 %	10.36 %	14.93 %	14.76 %	15.09 %	26.95 %	26.89 %	27.02 %	38.90 %	38.86 %	38.94 %
r.8	10.78 %	10.52 %	11.04 %	14.84 %	14.66 %	15.01 %	26.66 %	26.59 %	26.72 %	38.25 %	38.22 %	38.29 %
r.9	9.82 %	9.53 %	10.11 %	15.27 %	15.09 %	15.45 %	26.47 %	26.41 %	26.53 %	38.53 %	38.49 %	38.56 %
r.10	6.05 %	5.87 %	6.23 %	8.29 %	8.18 %	8.40 %	14.35 %	14.30 %	14.39 %	20.29 %	20.25 %	20.33 %
r.11	5.78 %	5.61 %	5.95 %	8.00 %	7.91 %	8.09 %	15.17 %	15.13 %	15.22 %	20.16 %	20.12 %	20.20 %
r.12	6.24 %	6.05 %	6.43 %	8.77 %	8.66 %	8.88 %	15.15 %	15.10 %	15.21 %	21.19 %	21.14 %	21.23 %
r.13	6.63 %	6.42 %	6.85 %	9.10 %	8.98 %	9.23 %	14.03 %	13.98 %	14.08 %	20.24 %	20.20 %	20.28 %
r.14	6.98 %	6.74 %	7.22 %	8.86 %	8.75 %	8.98 %	14.26 %	14.21 %	14.31 %	20.92 %	20.88 %	20.95 %
r.15	7.26 %	7.01 %	7.52 %	9.58 %	9.46 %	9.71 %	14.76 %	14.72 %	14.80 %	20.62 %	20.58 %	20.66 %
r.16	7.52 %	7.30 %	7.75 %	9.32 %	9.21 %	9.44 %	14.97 %	14.91 %	15.02 %	20.70 %	20.67 %	20.73 %
r.17	6.46 %	6.25 %	6.67 %	9.33 %	9.20 %	9.45 %	14.21 %	14.15 %	14.27 %	20.51 %	20.48 %	20.55 %
r.18	7.71 %	7.47 %	7.96 %	10.06 %	9.92 %	10.21 %	14.71 %	14.65 %	14.77 %	20.84 %	20.81 %	20.88 %

Table 37: Evaluations results: Trigger condition to send a new CAM in a scenario with 250 vehicles.

Run configuration	Mean Trigger 1	Lower confidence value	Upper confidence value	Mean Trigger 2	Lower confidence value	Upper confidence value	Mean Trigger 3	Lower confidence value	Upper confidence value	Mean Trigger 4	Lower confidence value	Upper confidence value
r.1	1.728 %	1.660 %	1.797 %	94.033 %	93.802 %	94.264 %	1.245 %	1.206 %	1.283 %	2.993 %	2.796 %	3.191 %
r.2	3.739 %	3.590 %	3.887 %	92.006 %	91.715 %	92.298 %	1.194 %	1.155 %	1.232 %	3.061 %	2.879 %	3.243 %
r.3	5.725 %	5.485 %	5.965 %	89.001 %	88.568 %	89.433 %	1.173 %	1.136 %	1.210 %	4.101 %	3.848 %	4.354 %
r.4	1.740 %	1.671 %	1.810 %	95.734 %	95.589 %	95.879 %	1.236 %	1.197 %	1.276 %	1.290 %	1.179 %	1.401 %
r.5	3.765 %	3.615 %	3.916 %	93.701 %	93.504 %	93.898 %	1.192 %	1.151 %	1.232 %	1.342 %	1.253 %	1.430 %
r.6	5.823 %	5.576 %	6.071 %	91.787 %	91.487 %	92.087 %	1.187 %	1.150 %	1.224 %	1.202 %	1.115 %	1.289 %
r.7	1.745 %	1.675 %	1.815 %	96.673 %	96.576 %	96.770 %	1.251 %	1.214 %	1.288 %	0.331 %	0.287 %	0.375 %
r.8	3.775 %	3.623 %	3.926 %	94.779 %	94.614 %	94.944 %	1.193 %	1.153 %	1.232 %	0.254 %	0.216 %	0.292 %
r.9	5.867 %	5.616 %	6.118 %	92.864 %	92.606 %	93.121 %	1.190 %	1.152 %	1.228 %	0.080 %	0.059 %	0.101 %
r.10	3.141 %	3.020 %	3.262 %	82.254 %	81.598 %	82.911 %	2.460 %	2.406 %	2.515 %	12.144 %	11.531 %	12.757 %
r.11	6.627 %	6.382 %	6.871 %	79.928 %	79.196 %	80.660 %	2.295 %	2.236 %	2.354 %	11.151 %	10.577 %	11.725 %
r.12	9.808 %	9.430 %	10.186 %	75.767 %	74.875 %	76.658 %	2.211 %	2.157 %	2.266 %	12.214 %	11.564 %	12.864 %
r.13	3.204 %	3.079 %	3.330 %	87.949 %	87.575 %	88.323 %	2.504 %	2.449 %	2.559 %	6.343 %	6.031 %	6.655 %
r.14	6.781 %	6.524 %	7.038 %	85.551 %	85.069 %	86.034 %	2.339 %	2.279 %	2.399 %	5.329 %	5.030 %	5.627 %
r.15	10.220 %	9.811 %	10.629 %	81.836 %	81.208 %	82.463 %	2.298 %	2.243 %	2.353 %	5.646 %	5.296 %	5.997 %
r.16	3.233 %	3.105 %	3.360 %	89.817 %	89.531 %	90.103 %	2.527 %	2.473 %	2.581 %	4.424 %	4.195 %	4.652 %
r.17	6.851 %	6.586 %	7.115 %	87.184 %	86.783 %	87.585 %	2.371 %	2.311 %	2.430 %	3.595 %	3.366 %	3.823 %
r.18	10.543 %	10.110 %	10.976 %	84.855 %	84.354 %	85.356 %	2.362 %	2.306 %	2.417 %	2.241 %	2.032 %	2.450 %

Table 38: Evaluations results: Trigger condition to send a new CAM in a scenario with 500 vehicles.

Run configuration	Mean Trigger 1	Lower confidence value	Upper confidence value	Mean Trigger 2	Lower confidence value	Upper confidence value	Mean Trigger 3	Lower confidence value	Upper confidence value	Mean Trigger 4	Lower confidence value	Upper confidence value
r.1	4.096 %	3.885 %	4.308 %	90.516 %	90.284 %	90.748 %	1.166 %	1.140 %	1.193 %	4.221 %	4.041 %	4.402 %
r.2	6.780 %	6.444 %	7.117 %	85.291 %	84.932 %	85.650 %	1.141 %	1.116 %	1.166 %	6.788 %	6.529 %	7.047 %
r.3	3.916 %	3.644 %	4.188 %	84.041 %	83.778 %	84.304 %	1.146 %	1.121 %	1.171 %	10.897 %	10.570 %	11.224 %
r.4	4.130 %	3.918 %	4.343 %	93.167 %	92.966 %	93.368 %	1.169 %	1.143 %	1.195 %	1.534 %	1.459 %	1.608 %
r.5	6.891 %	6.548 %	7.235 %	90.884 %	90.508 %	91.259 %	1.160 %	1.137 %	1.184 %	1.065 %	0.996 %	1.134 %
r.6	4.031 %	3.751 %	4.311 %	93.501 %	93.181 %	93.821 %	1.167 %	1.141 %	1.193 %	1.301 %	1.238 %	1.365 %
r.7	4.146 %	3.932 %	4.360 %	94.432 %	94.228 %	94.635 %	1.157 %	1.130 %	1.184 %	0.265 %	0.240 %	0.290 %
r.8	6.923 %	6.576 %	7.270 %	91.797 %	91.450 %	92.144 %	1.170 %	1.146 %	1.194 %	0.110 %	0.090 %	0.130 %
r.9	4.063 %	3.781 %	4.346 %	94.737 %	94.444 %	95.029 %	1.182 %	1.156 %	1.208 %	0.018 %	0.011 %	0.025 %
r.10	7.066 %	6.714 %	7.418 %	73.595 %	72.985 %	74.205 %	2.164 %	2.121 %	2.207 %	17.175 %	16.700 %	17.650 %
r.11	11.037 %	10.511 %	11.563 %	64.666 %	63.922 %	65.410 %	1.998 %	1.963 %	2.034 %	22.299 %	21.467 %	23.131 %
r.12	6.386 %	5.941 %	6.831 %	62.745 %	62.178 %	63.312 %	2.044 %	2.015 %	2.074 %	28.824 %	27.968 %	29.680 %
r.13	7.366 %	6.994 %	7.737 %	81.966 %	81.606 %	82.325 %	2.244 %	2.200 %	2.288 %	8.425 %	8.151 %	8.698 %
r.14	11.780 %	11.220 %	12.340 %	73.632 %	73.119 %	74.145 %	2.127 %	2.091 %	2.163 %	12.461 %	11.937 %	12.984 %
r.15	6.922 %	6.451 %	7.393 %	73.511 %	73.197 %	73.825 %	2.245 %	2.213 %	2.276 %	17.322 %	16.716 %	17.929 %
r.16	7.488 %	7.110 %	7.865 %	85.521 %	85.223 %	85.820 %	2.280 %	2.235 %	2.325 %	4.711 %	4.488 %	4.935 %
r.17	12.173 %	11.593 %	12.754 %	82.127 %	81.584 %	82.670 %	2.193 %	2.156 %	2.229 %	3.507 %	3.320 %	3.694 %
r.18	7.269 %	6.779 %	7.760 %	84.056 %	83.593 %	84.520 %	2.367 %	2.334 %	2.400 %	6.307 %	5.989 %	6.625 %

Table 39: Evaluations results: Trigger condition to send a new CAM in a scenario with 1000 vehicles.

Run configuration	Mean Trigger 1	Lower confidence value	Upper confidence value	Mean Trigger 2	Lower confidence value	Upper confidence value	Mean Trigger 3	Lower confidence value	Upper confidence value	Mean Trigger 4	Lower confidence value	Upper confidence value
r.1	2.996 %	2.889 %	3.103 %	94.846 %	94.656 %	95.035 %	0.797 %	0.777 %	0.816 %	1.362 %	1.271 %	1.453 %
r.2	1.389 %	1.335 %	1.443 %	95.376 %	95.225 %	95.528 %	0.801 %	0.781 %	0.821 %	2.433 %	2.314 %	2.552 %
r.3	1.158 %	1.126 %	1.190 %	95.051 %	94.904 %	95.199 %	0.809 %	0.789 %	0.830 %	2.981 %	2.859 %	3.103 %
r.4	3.012 %	2.904 %	3.120 %	95.687 %	95.543 %	95.832 %	0.786 %	0.766 %	0.806 %	0.515 %	0.477 %	0.553 %
r.5	1.399 %	1.344 %	1.454 %	97.622 %	97.548 %	97.695 %	0.794 %	0.773 %	0.814 %	0.186 %	0.169 %	0.202 %
r.6	1.168 %	1.136 %	1.201 %	97.484 %	97.422 %	97.546 %	0.801 %	0.780 %	0.821 %	0.547 %	0.522 %	0.571 %
r.7	3.023 %	2.914 %	3.132 %	96.057 %	95.930 %	96.183 %	0.789 %	0.769 %	0.809 %	0.131 %	0.116 %	0.146 %
r.8	1.400 %	1.345 %	1.454 %	97.799 %	97.732 %	97.865 %	0.797 %	0.777 %	0.817 %	0.005 %	0.002 %	0.007 %
r.9	1.174 %	1.142 %	1.207 %	98.012 %	97.966 %	98.058 %	0.813 %	0.792 %	0.834 %	0.000 %	0.000 %	0.001 %
r.10	5.172 %	4.999 %	5.345 %	69.853 %	69.321 %	70.386 %	1.521 %	1.489 %	1.552 %	23.454 %	23.011 %	23.897 %
r.11	2.354 %	2.266 %	2.442 %	66.148 %	65.654 %	66.642 %	1.502 %	1.471 %	1.533 %	29.996 %	29.545 %	30.447 %
r.12	1.957 %	1.905 %	2.009 %	65.447 %	64.948 %	65.946 %	1.510 %	1.479 %	1.541 %	31.086 %	30.627 %	31.545 %
r.13	5.525 %	5.337 %	5.714 %	89.109 %	88.815 %	89.402 %	1.626 %	1.592 %	1.659 %	3.740 %	3.616 %	3.864 %
r.14	2.579 %	2.481 %	2.676 %	87.637 %	87.371 %	87.903 %	1.626 %	1.591 %	1.661 %	8.159 %	7.962 %	8.355 %
r.15	2.150 %	2.092 %	2.209 %	87.194 %	86.924 %	87.463 %	1.646 %	1.611 %	1.681 %	9.010 %	8.793 %	9.227 %
r.16	5.559 %	5.367 %	5.750 %	91.635 %	91.385 %	91.885 %	1.632 %	1.598 %	1.666 %	1.174 %	1.107 %	1.242 %
r.17	2.615 %	2.515 %	2.714 %	95.181 %	95.043 %	95.320 %	1.661 %	1.626 %	1.696 %	0.543 %	0.505 %	0.580 %
r.18	2.205 %	2.144 %	2.265 %	95.806 %	95.700 %	95.913 %	1.678 %	1.642 %	1.715 %	0.310 %	0.269 %	0.351 %

Table 40: Evaluations results: Trigger condition to send a new CAM in a scenario with 1500 vehicles.

Run configuration	Mean Trigger 1	Lower confidence value	Upper confidence value	Mean Trigger 2	Lower confidence value	Upper confidence value	Mean Trigger 3	Lower confidence value	Upper confidence value	Mean Trigger 4	Lower confidence value	Upper confidence value
r.1	0.077 %	0.063 %	0.090 %	99.405 %	99.385 %	99.426 %	0.496 %	0.485 %	0.507 %	0.022 %	0.017 %	0.027 %
r.2	0.041 %	0.033 %	0.049 %	99.433 %	99.416 %	99.450 %	0.497 %	0.486 %	0.508 %	0.029 %	0.022 %	0.036 %
r.3	0.030 %	0.023 %	0.037 %	99.428 %	99.410 %	99.446 %	0.496 %	0.485 %	0.507 %	0.046 %	0.036 %	0.056 %
r.4	0.077 %	0.064 %	0.090 %	99.420 %	99.401 %	99.439 %	0.495 %	0.484 %	0.506 %	0.008 %	0.005 %	0.012 %
r.5	0.041 %	0.033 %	0.049 %	99.459 %	99.445 %	99.474 %	0.493 %	0.482 %	0.504 %	0.007 %	0.004 %	0.009 %
r.6	0.030 %	0.023 %	0.037 %	99.461 %	99.447 %	99.475 %	0.496 %	0.485 %	0.507 %	0.013 %	0.010 %	0.017 %
r.7	0.077 %	0.064 %	0.090 %	99.427 %	99.410 %	99.445 %	0.495 %	0.484 %	0.506 %	0.001 %	0.000 %	0.001 %
r.8	0.041 %	0.033 %	0.049 %	99.466 %	99.452 %	99.479 %	0.493 %	0.482 %	0.504 %	0.000 %	0.000 %	0.001 %
r.9	0.030 %	0.023 %	0.037 %	99.475 %	99.462 %	99.487 %	0.496 %	0.485 %	0.507 %	0.000 %	0.000 %	0.000 %
r.10	0.143 %	0.119 %	0.167 %	95.954 %	95.790 %	96.119 %	1.053 %	1.034 %	1.072 %	2.850 %	2.694 %	3.006 %
r.11	0.077 %	0.061 %	0.092 %	95.904 %	95.734 %	96.075 %	1.060 %	1.041 %	1.078 %	2.959 %	2.796 %	3.122 %
r.12	0.055 %	0.043 %	0.068 %	95.897 %	95.726 %	96.068 %	1.059 %	1.040 %	1.078 %	2.988 %	2.824 %	3.153 %
r.13	0.144 %	0.120 %	0.169 %	98.706 %	98.665 %	98.748 %	1.062 %	1.044 %	1.081 %	0.087 %	0.070 %	0.104 %
r.14	0.078 %	0.062 %	0.093 %	98.696 %	98.653 %	98.739 %	1.059 %	1.040 %	1.078 %	0.167 %	0.138 %	0.196 %
r.15	0.056 %	0.043 %	0.069 %	98.706 %	98.664 %	98.748 %	1.063 %	1.045 %	1.082 %	0.174 %	0.145 %	0.204 %
r.16	0.145 %	0.120 %	0.169 %	98.777 %	98.743 %	98.811 %	1.061 %	1.042 %	1.080 %	0.017 %	0.012 %	0.022 %
r.17	0.078 %	0.062 %	0.093 %	98.852 %	98.826 %	98.878 %	1.063 %	1.044 %	1.082 %	0.007 %	0.003 %	0.010 %
r.18	0.057 %	0.044 %	0.070 %	98.871 %	98.847 %	98.895 %	1.060 %	1.041 %	1.079 %	0.012 %	0.008 %	0.017 %

Table 41: Evaluations results: Average of the maximum perception range, for all different evaluation run configurations r.1 to r.18, each for the scenario with 250, 500, 1000 and 1500 vehicles.

Run configuration	Mean, 250 vehicles	Lower confidence value	Upper confidence value	Mean, 500 vehicles	Lower confidence value	Upper confidence value	Mean, 1000 vehicles	Lower confidence value	Upper confidence value	Mean, 1500 vehicles	Lower confidence value	Upper confidence value
r.1	1203.29 m	1201.41 m	1205.18 m	1530.71 m	1525.88 m	1535.55 m	1464.99 m	1462.39 m	1467.58 m	1288.13 m	1286.92 m	1289.33 m
r.2	1211.31 m	1208.89 m	1213.73 m	1436.39 m	1432.47 m	1440.32 m	1510.38 m	1507.14 m	1513.61 m	1284.16 m	1282.83 m	1285.49 m
r.3	1204.76 m	1202.54 m	1206.97 m	1501.96 m	1496.64 m	1507.29 m	1447.50 m	1444.60 m	1450.40 m	1291.86 m	1290.69 m	1293.02 m
r.4	1207.19 m	1205.05 m	1209.33 m	1553.05 m	1548.63 m	1557.47 m	1512.71 m	1509.20 m	1516.22 m	1287.03 m	1286.00 m	1288.06 m
r.5	1205.26 m	1203.09 m	1207.42 m	1513.60 m	1509.58 m	1517.61 m	1529.56 m	1526.09 m	1533.04 m	1317.19 m	1316.04 m	1318.33 m
r.6	1214.17 m	1212.14 m	1216.21 m	1536.06 m	1527.75 m	1544.38 m	1498.20 m	1494.84 m	1501.55 m	1282.23 m	1280.88 m	1283.57 m
r.7	1210.43 m	1208.03 m	1212.83 m	1506.82 m	1504.57 m	1509.08 m	1416.66 m	1414.45 m	1418.86 m	1286.23 m	1284.99 m	1287.48 m
r.8	1218.57 m	1215.64 m	1221.49 m	1569.82 m	1565.14 m	1574.51 m	1453.53 m	1451.19 m	1455.87 m	1274.40 m	1273.32 m	1275.48 m
r.9	1198.50 m	1196.16 m	1200.85 m	1525.83 m	1520.41 m	1531.25 m	1555.57 m	1550.88 m	1560.25 m	1315.42 m	1313.73 m	1317.10 m
r.10	1221.59 m	1219.43 m	1223.75 m	1452.55 m	1449.48 m	1455.61 m	1915.92 m	1912.46 m	1919.38 m	1786.88 m	1783.98 m	1789.77 m
r.11	1219.66 m	1217.64 m	1221.68 m	1503.62 m	1500.77 m	1506.48 m	1997.46 m	1994.43 m	2000.48 m	1855.08 m	1852.71 m	1857.45 m
r.12	1247.82 m	1244.32 m	1251.33 m	1579.19 m	1572.12 m	1586.26 m	1932.68 m	1929.33 m	1936.03 m	1766.39 m	1763.21 m	1769.57 m
r.13	1247.14 m	1244.21 m	1250.06 m	1609.57 m	1605.08 m	1614.07 m	2090.62 m	2087.98 m	2093.26 m	1727.89 m	1725.63 m	1730.14 m
r.14	1255.56 m	1252.92 m	1258.20 m	1611.41 m	1605.77 m	1617.05 m	1996.71 m	1994.70 m	1998.72 m	1833.43 m	1831.23 m	1835.63 m
r.15	1260.02 m	1257.07 m	1262.98 m	1636.27 m	1632.24 m	1640.29 m	2154.33 m	2150.12 m	2158.53 m	1874.64 m	1866.77 m	1882.50 m
r.16	1287.06 m	1283.64 m	1290.49 m	1673.13 m	1665.68 m	1680.58 m	2085.14 m	2082.85 m	2087.44 m	1826.02 m	1823.67 m	1828.37 m
r.17	1236.50 m	1234.27 m	1238.73 m	1711.26 m	1707.50 m	1715.03 m	2059.10 m	2055.05 m	2063.15 m	1750.18 m	1747.61 m	1752.76 m
r.18	1278.93 m	1276.10 m	1281.76 m	1674.82 m	1667.66 m	1681.98 m	2120.37 m	2118.32 m	2122.42 m	1847.50 m	1844.89 m	1850.10 m

Table 42: Evaluations results: Average completeness of local vehicle knowledge with respect to the ratio of known vehicles to existing vehicles in the respective distance, for a scenario with 250 vehicles.

Run configuration	Mean 250 m distance	Lower confidence value	Upper confidence value	Mean 500 m distance	Lower confidence value	Upper confidence value	Mean 750 m distance	Lower confidence value	Upper confidence value	Mean 1000 m distance	Lower confidence value	Upper confidence value
r.1	96.33 %	96.26 %	96.41 %	99.19 %	99.14 %	99.23 %	98.18 %	98.11 %	98.25 %	85.82 %	85.59 %	86.05 %
r.2	96.25 %	96.18 %	96.33 %	98.97 %	98.91 %	99.03 %	97.89 %	97.81 %	97.97 %	85.21 %	84.98 %	85.44 %
r.3	96.24 %	96.17 %	96.32 %	98.94 %	98.87 %	99.00 %	97.85 %	97.75 %	97.94 %	85.29 %	85.04 %	85.54 %
r.4	96.31 %	96.23 %	96.38 %	99.21 %	99.17 %	99.26 %	98.17 %	98.09 %	98.24 %	85.64 %	85.41 %	85.88 %
r.5	96.31 %	96.24 %	96.39 %	99.18 %	99.15 %	99.22 %	98.10 %	98.04 %	98.16 %	85.32 %	85.09 %	85.55 %
r.6	96.24 %	96.17 %	96.32 %	98.99 %	98.94 %	99.03 %	97.89 %	97.81 %	97.96 %	85.65 %	85.41 %	85.90 %
r.7	96.28 %	96.20 %	96.35 %	99.04 %	99.00 %	99.08 %	97.97 %	97.91 %	98.02 %	85.31 %	85.07 %	85.56 %
r.8	96.24 %	96.17 %	96.32 %	98.86 %	98.80 %	98.91 %	97.76 %	97.68 %	97.85 %	84.97 %	84.72 %	85.22 %
r.9	96.31 %	96.24 %	96.38 %	99.21 %	99.17 %	99.25 %	98.24 %	98.18 %	98.30 %	85.57 %	85.35 %	85.79 %
r.10	96.24 %	96.17 %	96.32 %	98.95 %	98.90 %	98.99 %	97.52 %	97.47 %	97.57 %	84.60 %	84.36 %	84.84 %
r.11	96.24 %	96.17 %	96.32 %	98.94 %	98.89 %	98.99 %	97.54 %	97.47 %	97.60 %	84.72 %	84.47 %	84.96 %
r.12	96.29 %	96.21 %	96.36 %	99.10 %	99.06 %	99.15 %	97.63 %	97.58 %	97.69 %	84.49 %	84.23 %	84.74 %
r.13	96.25 %	96.17 %	96.32 %	98.93 %	98.88 %	98.97 %	97.39 %	97.32 %	97.47 %	84.26 %	83.99 %	84.52 %
r.14	96.22 %	96.14 %	96.29 %	98.75 %	98.68 %	98.83 %	97.34 %	97.27 %	97.42 %	84.33 %	84.08 %	84.57 %
r.15	96.26 %	96.19 %	96.33 %	98.88 %	98.82 %	98.94 %	97.43 %	97.35 %	97.52 %	84.30 %	84.06 %	84.55 %
r.16	96.20 %	96.13 %	96.28 %	98.68 %	98.62 %	98.75 %	97.11 %	97.02 %	97.19 %	83.84 %	83.59 %	84.09 %
r.17	96.26 %	96.19 %	96.34 %	98.96 %	98.91 %	99.02 %	97.48 %	97.42 %	97.54 %	84.58 %	84.33 %	84.82 %
r.18	96.21 %	96.14 %	96.29 %	98.77 %	98.71 %	98.83 %	97.18 %	97.11 %	97.26 %	83.87 %	83.63 %	84.12 %

Table 43: Evaluations results: Average completeness of local vehicle knowledge with respect to the ratio of known vehicles to existing vehicles in the respective distance, for a scenario with 500 vehicles.

Run configuration	Mean 250 m distance	Lower confidence value	Upper confidence value	Mean 500 m distance	Lower confidence value	Upper confidence value	Mean 750 m distance	Lower confidence value	Upper confidence value	Mean 1000 m distance	Lower confidence value	Upper confidence value
r.1	97.73 %	97.70 %	97.75 %	98.24 %	98.21 %	98.27 %	97.08 %	97.05 %	97.12 %	90.18 %	97.05 %	97.12 %
r.2	97.82 %	97.80 %	97.84 %	98.51 %	98.49 %	98.54 %	97.42 %	97.39 %	97.44 %	90.41 %	97.39 %	97.44 %
r.3	97.84 %	97.82 %	97.85 %	98.51 %	98.48 %	98.54 %	97.36 %	97.32 %	97.39 %	90.18 %	97.32 %	97.39 %
r.4	97.79 %	97.77 %	97.82 %	98.32 %	98.29 %	98.35 %	96.98 %	96.95 %	97.02 %	90.06 %	96.95 %	97.02 %
r.5	97.77 %	97.75 %	97.79 %	98.38 %	98.35 %	98.41 %	97.20 %	97.16 %	97.23 %	90.25 %	97.16 %	97.23 %
r.6	97.77 %	97.74 %	97.79 %	98.29 %	98.26 %	98.32 %	96.97 %	96.92 %	97.01 %	89.96 %	96.92 %	97.01 %
r.7	97.77 %	97.75 %	97.79 %	98.31 %	98.29 %	98.33 %	97.11 %	97.08 %	97.14 %	90.17 %	97.08 %	97.14 %
r.8	97.79 %	97.77 %	97.81 %	98.33 %	98.30 %	98.35 %	97.06 %	97.01 %	97.10 %	90.14 %	97.01 %	97.10 %
r.9	97.75 %	97.73 %	97.77 %	98.19 %	98.16 %	98.22 %	96.87 %	96.84 %	96.91 %	89.97 %	96.84 %	96.91 %
r.10	97.63 %	97.61 %	97.65 %	98.04 %	98.01 %	98.07 %	96.40 %	96.36 %	96.44 %	89.75 %	96.36 %	96.44 %
r.11	97.72 %	97.70 %	97.74 %	98.26 %	98.24 %	98.29 %	96.65 %	96.61 %	96.68 %	89.83 %	96.61 %	96.68 %
r.12	97.69 %	97.67 %	97.70 %	98.18 %	98.15 %	98.20 %	96.57 %	96.53 %	96.60 %	89.62 %	96.53 %	96.60 %
r.13	97.55 %	97.53 %	97.57 %	97.75 %	97.71 %	97.79 %	96.04 %	95.99 %	96.09 %	89.25 %	95.99 %	96.09 %
r.14	97.58 %	97.57 %	97.60 %	97.84 %	97.80 %	97.87 %	96.13 %	96.09 %	96.17 %	89.37 %	96.09 %	96.17 %
r.15	97.54 %	97.52 %	97.56 %	97.70 %	97.67 %	97.74 %	95.97 %	95.93 %	96.01 %	89.11 %	95.93 %	96.01 %
r.16	97.58 %	97.56 %	97.60 %	97.78 %	97.75 %	97.82 %	96.10 %	96.07 %	96.14 %	89.47 %	96.07 %	96.14 %
r.17	97.59 %	97.58 %	97.61 %	97.64 %	97.61 %	97.67 %	95.77 %	95.72 %	95.81 %	89.06 %	95.72 %	95.81 %
r.18	97.50 %	97.48 %	97.52 %	97.61 %	97.56 %	97.65 %	95.83 %	95.78 %	95.89 %	88.96 %	95.78 %	95.89 %

Table 44: Evaluations results: Average completeness of local vehicle knowledge with respect to the ratio of known vehicles to existing vehicles in the respective distance, for a scenario with 1000 vehicles.

Run configuration	Mean 1000 m distance	Lower confidence value	Upper confidence value	Mean 500 m distance	Lower confidence value	Upper confidence value	Mean 750 m distance	Lower confidence value	Upper confidence value	Mean 1500 m distance	Lower confidence value	Upper confidence value
r.1	98.64 %	98.64 %	98.65 %	97.99 %	97.98 %	98.00 %	96.34 %	96.33 %	96.36 %	93.21 %	93.19 %	93.24 %
r.2	98.64 %	98.63 %	98.64 %	97.85 %	97.84 %	97.87 %	96.21 %	96.20 %	96.23 %	93.07 %	93.05 %	93.10 %
r.3	98.64 %	98.63 %	98.64 %	97.97 %	97.95 %	97.98 %	96.33 %	96.30 %	96.35 %	93.20 %	93.18 %	93.23 %
r.4	98.64 %	98.64 %	98.64 %	97.97 %	97.96 %	97.98 %	96.35 %	96.34 %	96.37 %	93.26 %	93.23 %	93.28 %
r.5	98.57 %	98.56 %	98.57 %	97.78 %	97.77 %	97.80 %	96.13 %	96.11 %	96.15 %	93.01 %	92.99 %	93.03 %
r.6	98.62 %	98.62 %	98.63 %	97.93 %	97.92 %	97.94 %	96.28 %	96.27 %	96.30 %	93.18 %	93.16 %	93.21 %
r.7	98.64 %	98.64 %	98.64 %	97.98 %	97.97 %	97.99 %	96.34 %	96.32 %	96.35 %	93.23 %	93.21 %	93.25 %
r.8	98.64 %	98.63 %	98.64 %	97.94 %	97.92 %	97.95 %	96.29 %	96.28 %	96.31 %	93.14 %	93.12 %	93.17 %
r.9	98.60 %	98.59 %	98.60 %	97.84 %	97.83 %	97.85 %	96.18 %	96.17 %	96.20 %	93.10 %	93.08 %	93.12 %
r.10	94.27 %	94.23 %	94.31 %	92.17 %	92.13 %	92.22 %	89.33 %	89.29 %	89.37 %	86.50 %	86.47 %	86.54 %
r.11	93.71 %	93.67 %	93.74 %	91.54 %	91.51 %	91.57 %	88.63 %	88.60 %	88.67 %	85.69 %	85.64 %	85.73 %
r.12	94.29 %	94.26 %	94.32 %	92.24 %	92.21 %	92.28 %	89.32 %	89.28 %	89.36 %	86.34 %	86.30 %	86.38 %
r.13	93.16 %	93.13 %	93.19 %	90.50 %	90.47 %	90.54 %	87.31 %	87.27 %	87.34 %	84.47 %	84.43 %	84.50 %
r.14	92.87 %	92.83 %	92.92 %	90.39 %	90.35 %	90.43 %	87.28 %	87.24 %	87.33 %	84.43 %	84.38 %	84.48 %
r.15	92.80 %	92.76 %	92.84 %	90.18 %	90.13 %	90.23 %	87.01 %	86.96 %	87.06 %	84.16 %	84.12 %	84.21 %
r.16	92.74 %	92.70 %	92.78 %	89.96 %	89.91 %	90.01 %	86.76 %	86.71 %	86.81 %	83.86 %	83.81 %	83.90 %
r.17	93.73 %	93.70 %	93.76 %	91.07 %	91.03 %	91.11 %	87.99 %	87.94 %	88.03 %	85.25 %	85.21 %	85.29 %
r.18	93.13 %	93.09 %	93.17 %	90.43 %	90.39 %	90.47 %	87.35 %	87.31 %	87.38 %	84.49 %	84.45 %	84.53 %

Table 45: Evaluations results: Average completeness of local vehicle knowledge with respect to the ratio of known vehicles to existing vehicles in the respective distance, for a scenario with 1500 vehicles.

Run configuration	Mean 250 m distance	Lower confidence value	Upper confidence value	Mean 500 m distance	Lower confidence value	Upper confidence value	Mean 750 m distance	Lower confidence value	Upper confidence value	Mean 1000 m distance	Lower confidence value	Upper confidence value
r.1	98.43 %	98.42 %	98.43 %	95.88 %	95.87 %	95.90 %	92.90 %	92.88 %	92.91 %	90.33 %	90.30 %	90.35 %
r.2	98.38 %	98.37 %	98.38 %	95.66 %	95.64 %	95.67 %	92.49 %	92.47 %	92.50 %	89.86 %	89.84 %	89.88 %
r.3	98.37 %	98.37 %	98.38 %	95.68 %	95.67 %	95.69 %	92.62 %	92.60 %	92.63 %	90.03 %	90.02 %	90.05 %
r.4	98.34 %	98.34 %	98.34 %	95.63 %	95.62 %	95.64 %	92.50 %	92.49 %	92.52 %	89.85 %	89.83 %	89.87 %
r.5	98.37 %	98.37 %	98.37 %	95.69 %	95.67 %	95.70 %	92.60 %	92.59 %	92.62 %	89.95 %	89.94 %	89.97 %
r.6	98.35 %	98.35 %	98.36 %	95.65 %	95.64 %	95.67 %	92.62 %	92.61 %	92.64 %	90.05 %	90.03 %	90.07 %
r.7	98.38 %	98.38 %	98.38 %	95.74 %	95.73 %	95.75 %	92.65 %	92.63 %	92.66 %	90.12 %	90.10 %	90.14 %
r.8	98.39 %	98.39 %	98.40 %	95.75 %	95.73 %	95.76 %	92.65 %	92.63 %	92.67 %	90.06 %	90.04 %	90.08 %
r.9	98.34 %	98.33 %	98.34 %	95.63 %	95.61 %	95.64 %	92.51 %	92.49 %	92.52 %	89.91 %	89.89 %	89.93 %
r.10	88.49 %	88.46 %	88.52 %	83.53 %	83.49 %	83.56 %	79.40 %	79.37 %	79.44 %	77.17 %	77.14 %	77.20 %
r.11	88.85 %	88.83 %	88.87 %	83.96 %	83.93 %	83.98 %	79.75 %	79.72 %	79.78 %	77.59 %	77.56 %	77.62 %
r.12	88.04 %	88.01 %	88.08 %	83.20 %	83.17 %	83.23 %	79.10 %	79.07 %	79.13 %	76.89 %	76.86 %	76.93 %
r.13	88.48 %	88.45 %	88.50 %	83.49 %	83.47 %	83.51 %	79.40 %	79.37 %	79.43 %	77.24 %	77.20 %	77.28 %
r.14	88.23 %	88.20 %	88.25 %	83.15 %	83.12 %	83.17 %	79.01 %	78.98 %	79.03 %	76.84 %	76.81 %	76.87 %
r.15	88.58 %	88.56 %	88.61 %	83.64 %	83.62 %	83.67 %	79.55 %	79.52 %	79.59 %	77.29 %	77.26 %	77.33 %
r.16	88.52 %	88.49 %	88.54 %	83.42 %	83.39 %	83.44 %	79.24 %	79.21 %	79.26 %	77.03 %	77.00 %	77.06 %
r.17	88.56 %	88.54 %	88.59 %	83.53 %	83.50 %	83.55 %	79.36 %	79.33 %	79.39 %	77.17 %	77.14 %	77.21 %
r.18	88.50 %	88.48 %	88.53 %	83.45 %	83.43 %	83.48 %	79.27 %	79.24 %	79.30 %	77.02 %	76.99 %	77.05 %

Table 46: Evaluations results: Average Channel Busy Ratio (CBR) with the use of CPMs, for all different evaluation run configurations r.1 to r.18, each for the scenario with 250, 500, 1000 and 1500 vehicles.

Run configuration	Mean 250 vehicles	Lower confidence value	Upper confidence value	Mean 500 vehicles	Lower confidence value	Upper confidence value	Mean 1000 vehicles	Lower confidence value	Upper confidence value	Mean 1500 vehicles	Lower confidence value	Upper confidence value
r.0	8.27 %	8.14 %	8.40 %	15.50 %	15.38 %	15.61 %	21.42 %	21.38 %	21.45 %	21.54 %	21.52 %	21.56 %
r.1	3.45 %	3.39 %	3.51 %	8.52 %	8.44 %	8.61 %	19.48 %	19.43 %	19.53 %	27.18 %	27.15 %	27.21 %
r.2	3.47 %	3.41 %	3.53 %	8.37 %	8.28 %	8.46 %	19.59 %	19.54 %	19.64 %	26.90 %	26.87 %	26.92 %
r.3	3.51 %	3.45 %	3.57 %	8.37 %	8.28 %	8.46 %	19.49 %	19.43 %	19.54 %	27.16 %	27.12 %	27.19 %
r.4	3.44 %	3.38 %	3.50 %	8.48 %	8.39 %	8.56 %	19.48 %	19.43 %	19.54 %	26.81 %	26.78 %	26.84 %
r.5	3.45 %	3.39 %	3.51 %	8.37 %	8.28 %	8.46 %	19.71 %	19.65 %	19.77 %	26.98 %	26.95 %	27.01 %
r.6	3.50 %	3.44 %	3.56 %	8.46 %	8.37 %	8.55 %	19.53 %	19.47 %	19.58 %	27.16 %	27.13 %	27.19 %
r.7	3.48 %	3.42 %	3.54 %	8.46 %	8.37 %	8.55 %	19.63 %	19.57 %	19.69 %	27.10 %	27.07 %	27.12 %
r.8	3.44 %	3.38 %	3.50 %	8.50 %	8.41 %	8.59 %	19.53 %	19.48 %	19.59 %	26.83 %	26.81 %	26.86 %
r.9	3.44 %	3.38 %	3.51 %	8.53 %	8.44 %	8.61 %	19.62 %	19.57 %	19.67 %	26.86 %	26.83 %	26.89 %
r.10	3.06 %	3.01 %	3.12 %	8.01 %	7.92 %	8.10 %	20.65 %	20.60 %	20.70 %	30.10 %	30.07 %	30.12 %
r.11	3.06 %	3.00 %	3.11 %	7.86 %	7.78 %	7.95 %	20.07 %	20.01 %	20.13 %	30.13 %	30.10 %	30.15 %
r.12	3.02 %	2.96 %	3.07 %	7.86 %	7.77 %	7.94 %	20.27 %	20.22 %	20.33 %	29.80 %	29.78 %	29.83 %
r.13	3.06 %	3.01 %	3.12 %	8.01 %	7.92 %	8.10 %	20.49 %	20.44 %	20.54 %	30.21 %	30.18 %	30.24 %
r.14	3.07 %	3.01 %	3.13 %	8.02 %	7.93 %	8.10 %	20.41 %	20.37 %	20.46 %	30.17 %	30.14 %	30.19 %
r.15	3.04 %	2.98 %	3.09 %	7.87 %	7.78 %	7.95 %	20.47 %	20.42 %	20.52 %	30.22 %	30.19 %	30.24 %
r.16	3.03 %	2.98 %	3.09 %	7.97 %	7.88 %	8.05 %	20.59 %	20.54 %	20.63 %	30.22 %	30.19 %	30.25 %
r.17	3.03 %	2.98 %	3.09 %	8.02 %	7.93 %	8.12 %	20.74 %	20.69 %	20.79 %	30.22 %	30.19 %	30.24 %
r.18	3.05 %	2.99 %	3.10 %	7.94 %	7.85 %	8.03 %	20.60 %	20.56 %	20.65 %	30.24 %	30.22 %	30.27 %

Table 47: Evaluations results: Average amount of known other vehicles in the local vehicle database in a scenario with 250 vehicles.

Run configuration	Mean, #, only CAM	Lower confidence value	Upper confidence value	Mean, #, with CPM	Lower confidence value	Upper confidence value
r.1	63.538	62.700	64.376	110.798	109.570	112.025
r.2	63.079	62.263	63.894	110.765	109.541	111.989
r.3	63.343	62.510	64.177	110.503	109.274	111.733
r.4	63.690	62.853	64.527	110.771	109.543	111.998
r.5	63.316	62.488	64.144	110.941	109.669	112.213
r.6	63.409	62.568	64.251	110.759	109.513	112.004
r.7	63.314	62.477	64.152	110.712	109.488	111.936
r.8	63.174	62.342	64.006	110.334	109.123	111.546
r.9	63.675	62.837	64.513	110.810	109.586	112.034
r.10	62.707	61.856	63.558	110.376	109.122	111.630
r.11	62.834	61.971	63.698	110.284	109.024	111.543
r.12	62.846	61.998	63.694	110.018	108.801	111.235
r.13	62.604	61.743	63.465	110.198	108.983	111.412
r.14	62.371	61.542	63.201	110.699	109.420	111.977
r.15	62.464	61.629	63.300	110.195	108.956	111.434
r.16	62.331	61.489	63.173	109.977	108.743	111.212
r.17	62.810	61.949	63.670	110.114	108.855	111.374
r.18	62.321	61.471	63.172	109.848	108.635	111.062

Table 48: Evaluations results: Average amount of known other vehicles in the local vehicle database in a scenario with 500 vehicles.

Run configuration	Mean, #, only CAM	Lower confidence value	Upper confidence value	Mean, #, with CPM	Lower confidence value	Upper confidence value
r.1	120.390	119.611	121.168	219.818	218.577	221.060
r.2	120.857	120.076	121.639	220.897	219.636	222.158
r.3	120.631	119.868	121.394	220.700	219.453	221.946
r.4	120.407	119.628	121.186	219.622	218.421	220.824
r.5	120.477	119.706	121.247	220.256	219.075	221.438
r.6	120.343	119.585	121.101	220.555	219.373	221.738
r.7	120.157	119.371	120.942	220.236	219.019	221.454
r.8	120.313	119.559	121.067	220.241	219.051	221.431
r.9	119.889	119.125	120.654	219.966	218.752	221.180
r.10	119.779	118.993	120.566	219.646	218.404	220.887
r.11	120.157	119.357	120.958	219.084	217.865	220.302
r.12	120.020	119.224	120.817	219.011	217.768	220.254
r.13	118.891	118.115	119.667	219.681	218.467	220.895
r.14	119.515	118.724	120.306	218.776	217.544	220.008
r.15	119.191	118.407	119.974	219.169	217.910	220.427
r.16	119.077	118.272	119.881	220.319	219.089	221.550
r.17	118.915	118.124	119.706	218.708	217.470	219.945
r.18	118.695	117.938	119.452	219.341	218.153	220.528

Table 49: Evaluations results: Average amount of known other vehicles in the local vehicle database in a scenario with 1000 vehicles.

Run configuration	Mean, #, only CAM	Lower confidence value	Upper confidence value	Mean, #, with CPM	Lower confidence value	Upper confidence value
r.1	233.166	232.761	233.571	455.666	454.705	456.627
r.2	232.096	231.712	232.481	455.677	454.748	456.607
r.3	233.040	232.658	233.423	455.837	454.919	456.755
r.4	232.885	232.502	233.268	456.494	455.561	457.426
r.5	231.426	231.048	231.804	454.995	454.028	455.962
r.6	232.634	232.237	233.030	455.404	454.434	456.373
r.7	232.828	232.448	233.208	456.236	455.290	457.182
r.8	232.101	231.712	232.490	455.554	454.609	456.498
r.9	232.216	231.824	232.608	455.852	454.903	456.802
r.10	234.680	234.260	235.100	432.204	431.291	433.116
r.11	234.493	234.088	234.897	429.797	428.815	430.779
r.12	235.437	235.023	235.850	432.291	431.385	433.198
r.13	230.391	229.995	230.786	425.827	424.911	426.742
r.14	230.302	229.901	230.702	424.410	423.490	425.329
r.15	230.185	229.783	230.587	423.958	423.063	424.854
r.16	228.665	228.285	229.045	422.322	421.434	423.209
r.17	229.368	228.990	229.746	428.114	427.249	428.978
r.18	228.952	228.566	229.338	425.481	424.640	426.322

Table 50: Evaluations results: Average amount of known other vehicles in the local vehicle database in a scenario with 1500 vehicles.

Run configuration	Mean, #, only CAM	Lower confidence value	Upper confidence value	Mean, #, with CPM	Lower confidence value	Upper confidence value
r.1	332.014	331.787	332.241	655.830	655.345	656.314
r.2	330.867	330.652	331.082	653.632	653.144	654.120
r.3	330.636	330.402	330.870	654.491	653.980	655.001
r.4	331.112	330.900	331.325	653.257	652.778	653.736
r.5	331.414	331.179	331.648	653.793	653.287	654.300
r.6	330.564	330.341	330.788	654.466	653.984	654.947
r.7	331.395	331.164	331.626	654.389	653.889	654.890
r.8	331.930	331.715	332.145	654.447	653.989	654.905
r.9	330.699	330.472	330.926	653.852	653.347	654.357
r.10	326.120	325.910	326.330	573.867	573.467	574.267
r.11	327.071	326.824	327.319	576.118	575.708	576.527
r.12	326.236	325.978	326.494	571.094	570.713	571.476
r.13	324.403	324.186	324.620	573.393	572.967	573.818
r.14	324.013	323.767	324.259	570.852	570.440	571.263
r.15	325.262	325.012	325.512	574.834	574.423	575.244
r.16	324.413	324.196	324.630	573.239	572.840	573.639
r.17	325.284	325.045	325.523	573.684	573.282	574.086
r.18	324.482	324.254	324.710	572.964	572.565	573.364

Table 51: Evaluations results: Average Packet Error Rate (PER) of our approach with and without the use of CPMs for all different evaluation run configurations r.1 to r.18, each for the scenario with 250, 500, 1000 and 1500 vehicles.

Run configuration	Mean 250 vehicles	Lower confidence value	Upper confidence value	Mean 500 vehicles	Lower confidence value	Upper confidence value	Mean 1000 vehicles	Lower confidence value	Upper confidence value	Mean 1500 vehicles	Lower confidence value	Upper confidence value
r.1	9.79 %	9.55 %	10.04 %	16.62 %	16.48 %	16.77 %	25.69 %	25.65 %	25.73 %	33.95 %	33.92 %	33.98 %
r.2	11.20 %	10.90 %	11.50 %	16.60 %	16.46 %	16.75 %	25.90 %	25.85 %	25.95 %	34.52 %	34.49 %	34.55 %
r.3	10.63 %	10.36 %	10.89 %	16.58 %	16.43 %	16.73 %	25.95 %	25.89 %	26.00 %	34.46 %	34.44 %	34.49 %
r.4	10.13 %	9.86 %	10.40 %	16.38 %	16.24 %	16.52 %	25.90 %	25.86 %	25.95 %	34.60 %	34.57 %	34.63 %
r.5	11.13 %	10.86 %	11.40 %	16.55 %	16.40 %	16.70 %	26.20 %	26.14 %	26.25 %	34.50 %	34.47 %	34.53 %
r.6	10.60 %	10.33 %	10.87 %	16.92 %	16.76 %	17.08 %	25.77 %	25.71 %	25.82 %	34.67 %	34.64 %	34.70 %
r.7	10.36 %	10.10 %	10.62 %	16.77 %	16.62 %	16.91 %	26.17 %	26.11 %	26.23 %	34.53 %	34.50 %	34.56 %
r.8	11.05 %	10.80 %	11.30 %	16.63 %	16.48 %	16.79 %	25.87 %	25.82 %	25.92 %	34.19 %	34.16 %	34.21 %
r.9	10.18 %	9.90 %	10.45 %	17.03 %	16.86 %	17.19 %	25.83 %	25.78 %	25.88 %	34.83 %	34.81 %	34.86 %
r.10	8.09 %	7.86 %	8.32 %	13.07 %	12.94 %	13.20 %	19.62 %	19.58 %	19.67 %	23.26 %	23.23 %	23.29 %
r.11	7.69 %	7.49 %	7.89 %	12.96 %	12.85 %	13.08 %	21.00 %	20.96 %	21.04 %	23.39 %	23.36 %	23.42 %
r.12	8.22 %	7.98 %	8.46 %	13.45 %	13.35 %	13.55 %	21.08 %	21.03 %	21.13 %	23.97 %	23.95 %	24.00 %
r.13	8.52 %	8.28 %	8.77 %	14.34 %	14.20 %	14.47 %	19.73 %	19.69 %	19.78 %	22.91 %	22.89 %	22.94 %
r.14	8.64 %	8.36 %	8.91 %	13.97 %	13.85 %	14.08 %	19.60 %	19.54 %	19.66 %	23.18 %	23.16 %	23.21 %
r.15	8.69 %	8.43 %	8.94 %	14.48 %	14.35 %	14.61 %	19.59 %	19.55 %	19.63 %	23.40 %	23.38 %	23.43 %
r.16	9.55 %	9.29 %	9.82 %	14.01 %	13.90 %	14.11 %	19.42 %	19.37 %	19.47 %	23.25 %	23.22 %	23.27 %
r.17	8.55 %	8.29 %	8.80 %	14.26 %	14.13 %	14.39 %	19.76 %	19.71 %	19.81 %	23.21 %	23.18 %	23.24 %
r.18	9.57 %	9.30 %	9.83 %	14.54 %	14.41 %	14.66 %	19.54 %	19.50 %	19.59 %	23.40 %	23.38 %	23.43 %

Table 52: Evaluations results: Average number of generated CAM messages for a scenario with 500 vehicles, with a trigger check interval of 100ms and 10ms.

Run configuration	Mean, #, 100ms	Lower confidence value	Upper confidence value	Mean, #, 10ms	Lower confidence value	Upper confidence value
r.1	341.55	341.12	341.97	410.38	409.55	411.21
r.2	347.82	347.07	348.57	410.14	409.35	410.94
r.3	348.82	348.20	349.44	410.14	409.35	410.94
r.4	338.61	338.18	339.04	372.24	371.88	372.61
r.5	342.80	342.10	343.50	371.96	371.67	372.25
r.6	339.28	338.75	339.81	371.96	371.67	372.25
r.7	337.44	337.03	337.85	359.80	359.58	360.01
r.8	341.71	341.08	342.35	359.50	359.41	359.59
r.9	336.82	336.31	337.33	359.50	359.41	359.59
r.10	195.80	195.06	196.53	272.29	271.48	273.10
r.11	211.79	210.78	212.80	272.06	271.27	272.84
r.12	216.13	215.55	216.70	272.06	271.27	272.84
r.13	188.58	188.08	189.08	222.17	221.66	222.67
r.14	197.69	196.83	198.56	221.88	221.43	222.34
r.15	196.75	196.18	197.32	221.88	221.43	222.34
r.16	185.57	185.13	186.02	200.14	199.75	200.53
r.17	191.47	190.65	192.29	199.85	199.53	200.17
r.18	186.43	185.79	187.07	199.85	199.53	200.17

Table 53: Evaluations results: Average CBR for a scenario with 500 vehicles, with a trigger check interval of 10ms.

Run configuration	Mean, only CAM	Lower confidence value	Upper confidence value	Mean, with CPM	Lower confidence value	Upper confidence value
r.1	2.56 %	2.54 %	2.58 %	7.11 %	7.05 %	7.17 %
r.2	2.56 %	2.54 %	2.58 %	7.09 %	7.02 %	7.15 %
r.3	2.56 %	2.54 %	2.58 %	7.12 %	7.06 %	7.18 %
r.4	2.43 %	2.41 %	2.45 %	6.99 %	6.93 %	7.04 %
r.5	2.43 %	2.41 %	2.45 %	7.06 %	7.01 %	7.12 %
r.6	2.43 %	2.41 %	2.45 %	7.01 %	6.96 %	7.07 %
r.7	2.24 %	2.23 %	2.26 %	8.03 %	7.94 %	8.11 %
r.8	2.24 %	2.23 %	2.26 %	8.06 %	7.97 %	8.14 %
r.9	2.24 %	2.23 %	2.26 %	8.06 %	7.97 %	8.15 %
r.10	1.57 %	1.56 %	1.58 %	7.99 %	7.92 %	8.06 %
r.11	1.57 %	1.56 %	1.58 %	8.02 %	7.95 %	8.10 %
r.12	1.57 %	1.56 %	1.58 %	7.97 %	7.89 %	8.04 %
r.13	1.35 %	1.34 %	1.37 %	8.00 %	7.92 %	8.08 %
r.14	1.35 %	1.34 %	1.37 %	8.13 %	8.06 %	8.21 %
r.15	1.35 %	1.34 %	1.37 %	8.14 %	8.07 %	8.22 %
r.16	1.27 %	1.26 %	1.28 %	8.31 %	8.21 %	8.40 %
r.17	1.27 %	1.26 %	1.28 %	8.26 %	8.16 %	8.36 %
r.18	1.27 %	1.26 %	1.28 %	8.18 %	8.09 %	8.27 %

Table 54: Evaluations results: Average PER for a scenario with 500 vehicles, with a trigger check interval of 10ms.

Run configuration	Mean, only CAM	Lower confidence value	Upper confidence value	Mean, with CPM	Lower confidence value	Upper confidence value
r.1	3.12 %	3.06 %	3.18 %	6.39 %	6.31 %	6.46 %
r.2	2.97 %	2.91 %	3.04 %	6.22 %	6.13 %	6.31 %
r.3	3.04 %	2.97 %	3.11 %	6.24 %	6.15 %	6.32 %
r.4	2.83 %	2.76 %	2.89 %	6.24 %	6.16 %	6.32 %
r.5	2.83 %	2.77 %	2.88 %	6.70 %	6.61 %	6.78 %
r.6	2.88 %	2.82 %	2.94 %	6.53 %	6.44 %	6.63 %
r.7	16.65 %	16.28 %	17.01 %	17.63 %	17.35 %	17.92 %
r.8	16.47 %	16.11 %	16.83 %	17.40 %	17.12 %	17.69 %
r.9	16.49 %	16.14 %	16.84 %	17.53 %	17.26 %	17.80 %
r.10	3.71 %	3.61 %	3.80 %	13.65 %	13.55 %	13.75 %
r.11	3.65 %	3.56 %	3.73 %	12.83 %	12.73 %	12.94 %
r.12	3.64 %	3.56 %	3.73 %	12.78 %	12.67 %	12.89 %
r.13	3.65 %	3.56 %	3.74 %	14.95 %	14.83 %	15.06 %
r.14	3.64 %	3.55 %	3.73 %	14.60 %	14.49 %	14.71 %
r.15	3.70 %	3.61 %	3.79 %	14.45 %	14.32 %	14.57 %
r.16	3.33 %	3.25 %	3.41 %	15.79 %	15.68 %	15.91 %
r.17	3.28 %	3.20 %	3.35 %	15.37 %	15.24 %	15.50 %
r.18	3.34 %	3.26 %	3.42 %	16.35 %	16.21 %	16.49 %

B

AUTHOR'S PUBLICATIONS

MAIN PUBLICATIONS

[1] D. Burgstahler, S. Bergsträßer, D. Böhnstedt, A. Reinhardt, A. Alhamoud, F. Englert, J. Schmitt, and R. Steinmetz. "CPaaS: Context Processing as a Service." In: *PIK - Praxis der Informationsverarbeitung und Kommunikation* 36 (Feb. 2013), p. 47. ISSN: 1865-8342.

[2] D. Burgstahler, U. Lampe, N. Richerzhagen, and R. Steinmetz. "Push vs. Pull: An Energy Perspective." In: *Proceedings of the 6th IEEE International Conference on Service Oriented Computing & Applications (SOCA 2013)*. IEEE. Dec. 2013, pp. 190–193. ISBN: 978-1-4799-2701-2.

[3] D. Burgstahler, F. Knapp, S. Zöller, T. Rückelt, and R. Steinmetz. "Where is That Car Parked? A Wireless Sensor Network-Based Approach to Detect Car Positions." In: *Proceedings of the 9th IEEE LCN International Workshop on Practical Issues in Building Sensor Network Applications (IEEE SenseApp 2014)*. IEEE. Sept. 2014, pp. 514–522. ISBN: 978-1-4799-3782-0.

[4] D. Burgstahler, N. Richerzhagen, F. Englert, R. Hans, and R. Steinmetz. "Switching Push - Pull: An Energy Efficient Notification Approach." In: *Proceedings of the 3rd International Conference on Mobile Services (MS 2014)*. IEEE. June 2014, pp. 68–75. ISBN: 978-1-4799-5059-1.

[5] D. Burgstahler, S. Schulte, S. Abels, K. Kipp, P. Hoenisch, S. Dustdar, and R. Steinmetz. "Informationssysteme für Verkehrsteilnehmer: Datenintegration, Cloud-Dienste und der Persönliche Mobilitätsassistent." In: *PIK - Praxis der Informationsverarbeitung und Kommunikation* 37.3 (Sept. 2014), pp. 243–250. ISSN: 0930-5157.

[6] D. Burgstahler, M. Pelzer, A. Lotz, F. Knapp, H. Pu, T. Rückelt, and R. Steinmetz. "A Concept for a C2X-based Crossroad Assistant." In: *Proceedings of the 2nd IEEE PerCom Workshop on Smart Environments: Closing the Loop (SmartE 2015)*. IEEE. Mar. 2015. ISBN: 978-1-4799-8425-1.

[7] D. Burgstahler, S. Zöller, M. Möbus, T. Walter, T. Rückelt, and R. Steinmetz. "Navigate.KOM: Datenbankbasierter Informationsansatz für Fahrassistenzsysteme." In: *GMM-Fachbericht-AmE 2015–Automotive meets Electronics*. VDE VERLAG GmbH, Mar. 2015, pp. 111–116. ISBN: 978-3-8007-3890-8.

[8] D. Burgstahler, A. Xhoga, C. Peusens, M. Möbus, D. Böhnstedt, and R. Steinmetz. "RemoteHorizon.KOM: Dynamic Cloud-based eHorizon." In: *GMM-Fachbericht-AmE 2016–Automotive meets Electronics*. VDE VERLAG GmbH, Mar. 2016, pp. 46–51. ISBN: 978-3-8007-4167-0.

[9] D. Burgstahler, C. Peusens, D. Boehnstedt, and R. Steinmetz. "Horizon.KOM: A First Step Towards an Open Vehicular Horizon Provider." In: *Proceedings of the 2nd International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS)*. Apr. 2016, pp. 79–84. ISBN: 978-989-758-185-4.

[10] D. Burgstahler, T. Meuser, U. Lampe, D. Böhnstedt, and R. Steinmetz. "ProbSense.KOM: A Probabilistic Sensing Approach for Gathering Vehicular Sensed Data." In: *Proceedings of the 3rd International Conference on Mobile Services (MS 2016)*. IEEE. June 2016, pp. 9–16. ISBN: 978-1-5090-2625-8.

[11] D. Burgstahler, M. Möbus, T. Meuser, D. Böhnstedt, and R. Steinmetz. "A Categorization Scheme for Information Demands of Future Connected ADAS (accepted for publication)." In: *GMM-Fachbericht-AmE 2017–Automotive meets Electronics*. VDE VERLAG GmbH, Mar. 2017.

CO-AUTHORED PUBLICATIONS

[12] A. Reinhardt, P. Baumann, D. Burgstahler, M. Hollick, H. Chonov, M. Werner, and R. Steinmetz. "On the Accuracy of Appliance Identification Based on Distributed Load Metering Data." In: *Proceedings of the 2nd IFIP Conference on Sustainable Internet and ICT for Sustainability (SustainIT)*. Ed. by E. Ancillotti. IEEE Catalog Number CFP1217T-USB. IFIP, Oct. 2012, pp. 1–9. ISBN: 978-3-901882-46-3.

[13] A. Reinhardt and D. Burgstahler. "Exploiting Platform Heterogeneity in Wireless Sensor Networks by Shifting Resource-Intensive Tasks to Dedicated Processing Nodes." In: *Proceedings of the 14th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*. IEEE. June 2013, pp. 1–9. ISBN: 978-1-4673-5828-6.

[14] A. Alhamoud, S. Bergsträßer, D. Burgstahler, D. Böhnstedt, F. Englert, A. Reinhardt, J. Schmitt, and R. Steinmetz. *CPaaS: Context Processing as a Service*. Software Demonstration, NetSys Communication Software Award. Mar. 2013.

[15] F. Englert, A. El'Hindi, D. Burgstahler, A. Alhamoud, and R. Steinmetz. "Reducing the Electricity Consumption of Large Outdoor LED Advertising Screens." In: *Proceedings of the 5th International Conference on Future Energy Systems (ACM e-Energy)*. Ed. by ACM. ACM. June 2014, pp. 209–210. ISBN: 978-1-4503-2819-7.

[16] A. Alhamoud, M. Kreger, H. Afifi, C. Gottron, D. Burgstahler, F. Englert, D. Böhnstedt, and R. Steinmetz. "Empirical Investigation of the Effect of the Door's State on Received Signal Strength in Indoor Environments At 2.4 GHz." In: *Proceedings of the 10th IEEE International Workshop on Performance and Management of Wireless and Mobile Networks (IEEE P2MNET 2014)*. IEEE. Sept. 2014, pp. 652–657. ISBN: 978-1-4799-3782-0.

[17] R. Hans, U. Lampe, D. Burgstahler, M. Hellwig, and R. Steinmetz. "Where Did My Battery Go? Quantifying the Energy Consumption of Cloud Gaming." In: *Proceedings of the 3rd International Conference on Mobile Services (MS 2014)*. IEEE. June 2014, pp. 63–67. ISBN: 978-1-4799-5059-1.

[18] A. Alhamoud, F. Rüttiger, A. Reinhardt, F. Englert, D. Burgstahler, D. Böhnstedt, C. Gottron, and R. Steinmetz. "SMARTENERGY.KOM: An Intelligent System for Energy Saving in Smart Home." In: *Proceedings of the 3rd IEEE LCN International Workshop on Global Trends in SMART Cities (IEEE goSMART 2014)*. IEEE. Sept. 2014, pp. 685–692. ISBN: 978-1-4799-3782-0.

[19] T. Rückelt, D. Burgstahler, F. Englert, C. Gottron, S. Zöller, and R. Steinmetz. "A Concept for Vehicle Internet Connectivity for Non-Safety Applications." In: *Proceedings of the 8th IEEE LCN International Workshop On User MObility and VEhicular Networks (IEEE ON-MOVE 2014)*. IEEE. Sept. 2014. ISBN: 978-1-4799-3782-0.

[20] R. Hans, D. Burgstahler, A. Mueller, M. Zahn, and D. Stingl. "Knowledge for a Longer Life: Development Impetus for Energy-efficient Smartphone Applications." In: *Proceedings of the 4th International Conference on Mobile Services (MS 2015)*. IEEE. June 2015, pp. 128–133. ISBN: 978-1-4673-7283-1.

[21] T. Rueckelt, F. Jomrich, D. Burgstahler, D. Böhnstedt, and R. Steinmetz. "Publish-Subscribe-Based Control Mechanism for Scheduling Integration in Mobile IPv6." In: *Proceedings of the 40th IEEE Conference on the Local Computer Networks (LCN)*. IEEE. Oct. 2015, pp. 478–481. ISBN: 978-1-4673-6770-7.

[22] P. Lieser, F. Englert, A. Alhamoud, D. Burgstahler, and D. Boehnstedt. "Towards Virtual Sensors for Estimating the Electricity Consumption of Networked Appliances." In: *Proceedings of the 6th ACM International Conference on Future Energy Systems*. ACM. July 2015, pp. 205–206. ISBN: 978-1-4503-3609-3.

[23] S. Schulte, P. Hoenisch, K. Kipp, D. Burgstahler, S. Abels, and G. Liguori. "A Service Framework for Smart Mobility Scenarios." In: *Proceedings of the 3rd International Conference on Mobile Services (MS 2016)*. IEEE. June 2016, pp. 17–24. ISBN: 978-1-5090-2625-8.

[24] T. Rueckelt, D. Burgstahler, F. Jomrich, D. Böhnstedt, and R. Steinmetz. "Impact of Time in Network Selection for Mobile Nodes." In: *Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM 16)*. ACM. Nov. 2016, pp. 68–77. ISBN: 978-1-4503-4502-6.

[25] T. Rueckelt, H. Altug, D. Burgstahler, D. Böhnstedt, and R. Steinmetz. "MoVeNet: Mobility Management for Vehicular Networking." In: *Proceedings of the 14th ACM International Symposium on Mobility Management and Wireless Access (MobiWac 16)*. ACM. Nov. 2016, pp. 139–146. ISBN: 978-1-4503-4503-3.

[26] T. Meuser, D. Burgstahler, T. Rückelt, D. Böhnstedt, and R. Steinmetz. "Hybrid-ProbSense.KOM: Probabilistic Sensing with Hybrid Communication for Gathering Vehicular Sensed Data (accepted for publication)." In: *GMM-Fachbericht-AmE 2017–Automotive meets Electronics*. VDE VERLAG GmbH, Mar. 2017.

All previously listed publications have either been published or accepted for publication as of February 14, 2017.

C

CURRICULUM VITÆ

PERSONAL INFORMATION

Name	Daniel-Maurice Burgstahler
Date of Birth	May 23, 1980
Place of Birth	Wiesbaden, Germany
Nationality	German

EDUCATION

11/2010–11/2012	Technische Universität Darmstadt Darmstadt, Germany Department of Electrical Engineering Master of Science, *Elektrotechnik und Informationstechnik*
10/2002–10/2010	Technische Universität Darmstadt Darmstadt, Germany Department of Electrical Engineering Bachelor of Science, *Informations- und Kommunikaionstechnik*
10/2001–09/2002	Bingen University of Applied Sciences Bingen, Germany Department of Mechanical Engineering
9/2000–06/2001	Military Service, German Army Base Speyer Speyer, Germany Heavy engineering battalion, Lance Corporal
07/1990–06/2000	Sebastian Münster Gymnasium Ingelheim am Rhein, Germany Abitur, majoring in Mathematics, Physics and Geography

PROFESSIONAL EXPERIENCE

since 03/2017	Daimler AG Stuttgart, Germany Engineer Cloud and IoT, Future Transportation
12/2012-02/2017	Technische Universität Darmstadt Darmstadt, Germany Electrical Engineering & Information Technology Dept. Multimedia Communications Lab (KOM) Research assistant
01/2007–12/2010	Freelancer Software Development .NET Software development for Boehringer Ingelheim Pharma GmbH

	& Co. KG: Development of a generic FM data data management system with automatic CAD visualization
08/2002–09/2002	BFE Studio und Medien Systeme GmbH (internship) Mainz, Germany Basic training in mechanics and electronics
07/2001–09/2001	Boehringer Ingelheim Pharma GmbH & Co. KG (internship) Ingelheim am Rhein, Germany CAD documentation of technical facilities

SELECTED TEACHING ACTIVITIES

2013–2017	Tutor in Lab exercise Multimedia Communications Lab II
2014–2017	Tutor in Seminar Advanced Topics in Future Internet Research

SCIENTIFIC ACTIVITIES

Reviewer	IEEE Vehicular Technology Conference (VTC-Fall), 2013
	IEEE International Conference on Network Protocols (ICNP), 2013 (conjointly with Prof. R. Steinmetz)
	IEEE Conference on Local Computer Networks (LCN), 2014
	Hindawi Journal of Applied Mathematics, 2014 (conjointly with Prof. R. Steinmetz)
	IEEE International Conference on Internet-of-Things Design and Implementation (IoTDI), 2016 (conjointly with Prof. R. Steinmetz)
	ACM International Conference on Systems for Energy-Efficient Built Environments (BuildSys), 2016 (conjointly with Prof. R. Steinmetz)

SUPERVISED THESES

KOM-M-0555	C. Peusens, "Enrichment of a Connected Vehicular eHorizon by Detection of Driving Behaviour and Traffic Flow Anomalies", (conjointly with Dr. Eric Wahl, Dr. Ing. h.c. F. Porsche AG, Weissach). Master Thesis, Technische Universität Darmstadt, December 2016.
KOM-M-0558	A. El-Hindi, "Detection of Changes in Road Maps Based on Vehicular Sensors with Efficient Information Distribution", (conjointly with Dr. Michael Henzler, Daimler AG, Sindelfingen). Master Thesis, Technische Universität Darmstadt, November 2016.
KOM-M-0542	H. Min, "Variations in Server-based eHorizon Generation with Adaptive Transmission". Master Thesis, Technische Universität Darmstadt, October 2016.

KOM-M-0538 L. Backert, "Increased C2C Perception Range: Using Trajectory Prediction Filters to Reduce ITS G5 Channel Utilization and Increase Perception Range". Master Thesis, Technische Universität Darmstadt, June 2016.

KOM-M-0536 V. Hilsendeger, "Automatic Validation and Integration of Crowdsource Maps", (conjointly with Maximilian Muffert, Mercedes-Benz R. & D. North America, Inc., Sunnyvale). Master Thesis, Technische Universität Darmstadt, June 2016.

KOM-B-0544 J. Neureither, "Security and Privacy in Advanced Vehicular Data Collection Scenarios". Bachelor Thesis, Technische Universität Darmstadt, March 2016.

KOM-M-0526 M. Holz, "Messaging-based Geocasts: A Publish Subscribe based Geo-Messaging Approach". Master Thesis, Technische Universität Darmstadt, December 2015.

KOM-M-0525 T. Meuser, "Data Collection Management: A Dynamic Approach for Map Information Harvesting". Master Thesis, Technische Universität Darmstadt, December 2015.

KOM-M-0518 A. Xhoga, "Server-based eHorizon Generation: Collaborative Data Collection for an Extended Vehicle Information View" (conjointly with Michael Wagner, Adam Opel AG, Rüsselsheim). Master Thesis, Technische Universität Darmstadt, October 2015.

KOM-M-0508 F. Jomrich, "Seamless Handover in Wireless Vehicular Communication Networks", (conjointly with Tobias Rückelt). Master Thesis, Technische Universität Darmstadt, June 2015.

KOM-B-0513 S. Ikehata, "Optimizing the Usage of Electric Vehicles - Development of a Live Data-based Scheduler" (conjointly with Dr.-Ing. Sebastian Zöller and Dr.-Ing. Fernando Lyardet, The Uraban Institut). Seminar paper, Technische Universität Darmstadt, April 2015.

KOM-B-0513 M. Möbus, "Collaborative Data Collection: Development of a Data Format for Collecting Road Properties" (conjointly with Michael Wagner, Adam Opel AG, Rüsselsheim). Bachelor Thesis, Technische Universität Darmstadt, January 2015.

KOM-M-0496 M. Pelzer, "Development of a C2X-based Crossroad Assistant", (conjointly with Dr.-Ing. Hongjun Pu, Continental Automotive GmbH, Wezlar). Master Thesis, Technische Universität Darmstadt, August 2014.

KOM-B-0487 B. Börngen-Schmidt, "Development of a Generic Sensor Abstraction Layer for Vehicular Sensors". Seminar paper, Technische Universität Darmstadt, July 2014.

KOM-B-0485 F. Knapp, "Development of a Smart Parking Sensor and a Method to Detect Vehicle Positions on Parking Lots". Bachelor Thesis, Technische Universität Darmstadt, May 2014.

KOM-M-0489 J. Ngepi, "Energy Efficient Classification in Wireless Sensor Networks by the Use of Machine Learning Techniques". Master Thesis, Technische Universität Darmstadt, April 2014.

KOM-B-0484 C. Peusens, "Contextual Prediction of Road Properties as Location Based Service", Beste Bachelorarbeit 2014. Bachelor Thesis, Technische Universität Darmstadt, February 2014.

KOM-B-0481 M. T. Tomasi, "A Practical Evaluation of the Energy Demand of State-of-the-Art Point-to-Point Transport Protocols in WSNs". Bachelor Thesis, Technische Universität Darmstadt, January 2014.

KOM-M-0483 P. Chrzastowski, "Development of an Android-based recommendation system for optimizing drivers' behaviours using in-car sensors" (conjointly with Dr.-Ing. Sebastian Zöller). Master Thesis, Technische Universität Darmstadt, December 2013.

KOM-B-0476 A. El-Hindi: "Adaptive light control of a LED advertising sign based on outdoor luminance" (conjointly with Frank Englert). Bachelor Thesis, Technische Universität Darmstadt, October 2013.

D

ERKLÄRUNG LAUT §9 DER PROMOTIONSORDNUNG

Ich versichere hiermit, dass ich die vorliegende Dissertation allein und nur unter Verwendung der angegebenen Literatur verfasst habe.

Die Arbeit hat bisher noch nicht zu Prüfungszwecken gedient.

Darmstadt, 14. February 2017

Daniel-Maurice Burgstahler